UNIVERSITÀ DEGLI STUDI DI MILANO

DOTTORATO DI RICERCA IN INFORMATICA — XXXIV CICLO
DIPARTIMENTO DI INFORMATICA "GIOVANNI DEGLI ANTONI"

Design and Explainability of Machine Learning Algorithms for the Classification of Cardiac Abnormalities from Electrocardiogram Signals

INF/01

Doctoral Dissertation of:
Matteo Bodini

Supervisor:
Prof. Roberto Sassi

Co-Supervisor:
Dr. Massimo Walter Rivolta

Doctorate School's Director:
Prof. Paolo Boldi

Academic Year 2020 – 2021

Abstract

The research activity contained in the present thesis work is devoted to the development of novel Machine Learning (ML) and Deep Learning (DL) algorithms for the classification of Cardiac Abnormalities (CA) from Electrocardiogram (ECG) signals, along with the explanation of classification outputs with explainable approaches. Automated computer programs for ECG classification have been developed since 1950s to improve the correct interpretation of the ECG, nowadays facilitating health care decision-making by reducing costs and human errors. The first ECG interpretation computer programs were essentially developed by *translating into the machine* the domain knowledge provided by expert physicians. However, in the last years leading research groups proposed to employ standard ML algorithms (which involve feature extraction, followed by classification), and more recently *end-to-end* DL algorithms to build automated ECG classification computer programs for the detection of CA. Recently, several research works proposed DL algorithms which even exceeded the performance of board-certified cardiologists in detecting a wide range of CA from ECGs. As a matter of fact, DL algorithms seem to represent promising tools for automated ECG classification on the analyzed datasets. However, the latest research related to ML and DL carries two main drawbacks that were tackled throughout the doctoral experience. First, to let the standard ML algorithms to perform at their best, the proper preprocessing, feature engineering, and classification algorithm (along with its parameters and hyperparameters) must be selected. Even when end-to-end DL approaches are adopted, and the feature extraction step is automatically learned from data, the optimal model architecture is crucial to get the best performance. To address this issue, we exploited the domain knowledge of electrocardiography to design an ensemble ML classification algorithm to classify within a wide range of 27 CA. Differently from other works in the context of ECG classification, which often borrowed ML and DL architectures from other domains, we designed each model in the ensemble according to the domain knowledge to specifically classify a subset of the considered CA that alter the same set of ECG physiological features known by physicians. Furthermore, in a subsequent work, toward the same aim we experimented three different Automated ML frameworks to automatically find the optimal ML pipeline in the case of standard and end-to-end DL algorithms. Second, while several research articles reported

remarkable results for the value of ML and DL in classifying ECGs, only a handful offer insights into the model's learning representation of the ECG for the respective task. Without explaining what these models are sensing on the ECG to perform their classifications in an explainable way, the developers of such algorithms run a strong risk of discouraging the physicians to adopt these tools, since they need to understand how ML and DL work before entrusting it to facilitate their clinical practice. Methods to open the *black-boxes* of ML and DL have been applied to the ECG in a few works, but they often provided only explanations restricted to a single ECG at time and with limited, or even absent, framing into the knowledge domain of electrocardiography. To tackle such issues, we developed techniques to unveil which portions of the ECG were the most relevant to the classification output of a ML algorithm, by computing average explanations over all the training samples, and translating them for the physicians' understanding. In a preliminary work, we relied on the Local Interpretable Model-agnostic Explanations (LIME) explainability algorithm to highlight which ECG leads were the most relevant in the classification of ST-Elevation Myocardial Infarction with a Random Forest classifier. Then, in a subsequent work, we extended the approach and we designed two model-specific explainability algorithms for Convolutional Neural Networks to explain which ECG waves, a concept understood by physicians, were the most relevant in the classification process of a wide set of 27 CA for a state-of-the-art CNN.

Contents

List of Figures

List of Tables

1 Introduction

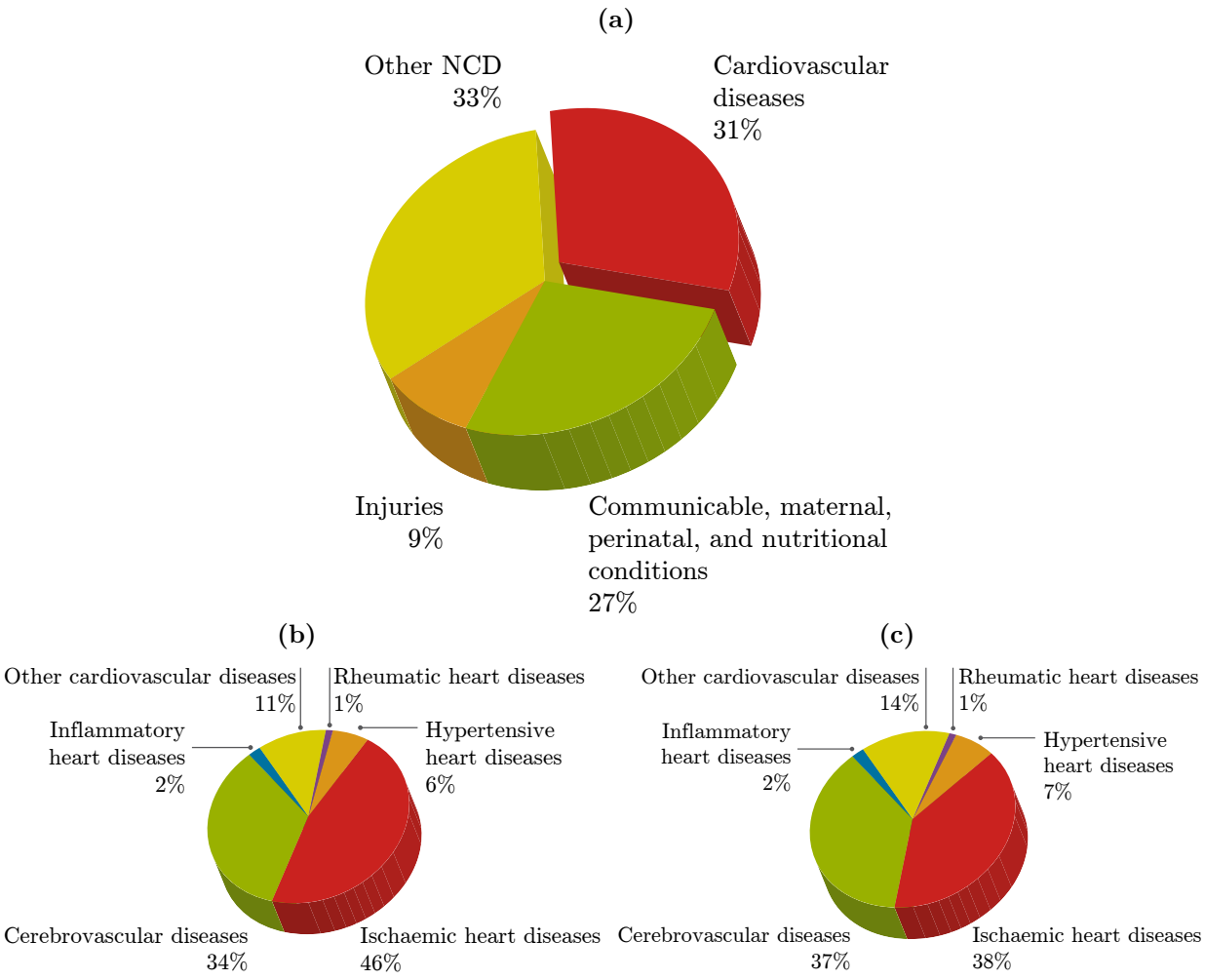
Contents

1.1 Cardiac Abnormalities

Cardiac Abnormalities (CA) fall under the umbrella of Cardiovascular Diseases (CVD) which include disorders that affect the heart muscle, the brain, and blood vessels [1]. CVD stand as the leading cause of mortality worldwide according to the statistics of the World Health Organization (WHO) [2]: more than 17.3 million of deaths per year are caused by CVD, which corresponds to more than 30% of global deaths, as reported in Figure 1.1(a) which shows the yearly distribution of leading causes of mortality worldwide.

The Figures 1.1(b) and 1.1(c) which report the yearly percentage of CVD deaths caused by different kinds of CVD, respectively for males and females, show that

Figure 1.1: (a) Distribution of the leading causes of mortality worldwide per year. The acronym NCD stands for "noncommunicable diseases". The yearly distribution of CVD deaths caused by several types of CVD are respectively reported in panel (b) and (c) for males and females. Statistics collected from the WHO [2].



more than a half of the yearly total deaths caused by CVD is related to CA. Different kinds of CA may be caused by several conditions mainly including atherosclerosis, congenital malformations, rheumatic fever, disorders of the heart muscles, and disorders of the electrical conduction system of the heart [1, 2].

Atherosclerosis is a pathological condition of blood vessels that may results in ischemic heart diseases or coronary artery diseases, usually known as *heart attacks* (while in the case of cerebrovascular diseases it may cause what is usually known as *stroke*) [3]. During atherosclerosis, fatty material and cholesterol are deposited inside

blood vessels, and such deposits[1] may cause the interior surface of the blood vessels to become irregular and their *lumens*[2] to become narrow, making it more difficult for the blood to flow through. Blood vessels also become less flexible as a direct result. Eventually, plaques may rupture, thus leading to the formation of a blood clot, and if such event happens in a coronary artery it may lead to a heart attack (while if it happens in the brain, it may lead to a stroke). Atherosclerosis is responsible for a large percentage of CVD: for instance, in 2008 heart attacks were responsible for 7.3 million deaths and strokes were responsible for 6.2 million deaths, out of the 17.3 million total CVD deaths [2]. Nowadays, there is strong scientific evidence that several risk factors promote the condition of atherosclerosis, including behavioral risk factors (tobacco smoking, an unhealthy diet, harmful use of alcohol, *etc.*), metabolic risk factors (hypertension, diabetes, obesity, *etc.*), and other risk factors (advancing age, genetic disposition, and psychological factors such as stress) [3].

Malformations of the structure of the heart muscle immediately noticeable at birth are known as *congenital heart malformations*. Common examples include holes in the heart septum, abnormal valves, deformations in heart chambers, *etc.* [5]. The arising of congenital heart malformations may be caused by a consanguinity relation between parents, maternal infections (*e.g.* the well-known Rubella, an infection caused by the Rubella virus [6]), maternal harmful use of alcohol or/and drugs, and poor maternal nutrition (*e.g.* lack of folic acid in the diet of people which live in less developed countries) [7].

Rheumatic heart disease is caused by damages to the heart muscle and heart valves caused by rheumatic fever, an inflammatory disease that can affect several connective tissues especially in the heart, joints, skin, or in the brain, following a streptococcal pharyngitis or tonsillitis [8]. The heart valves can be inflamed, thus becoming scarred over time, and their narrowing or leaking makes it harder for the heart to function normally. Such condition may take years to develop and it may

[1] Deposits composed of fatty material are usually referred as *plaques* [3].

[2] The term "lumen" here refers to the interior part of a vessel, *i.e.* the central space in an artery, vein, or capillary through which the blood flows [4].

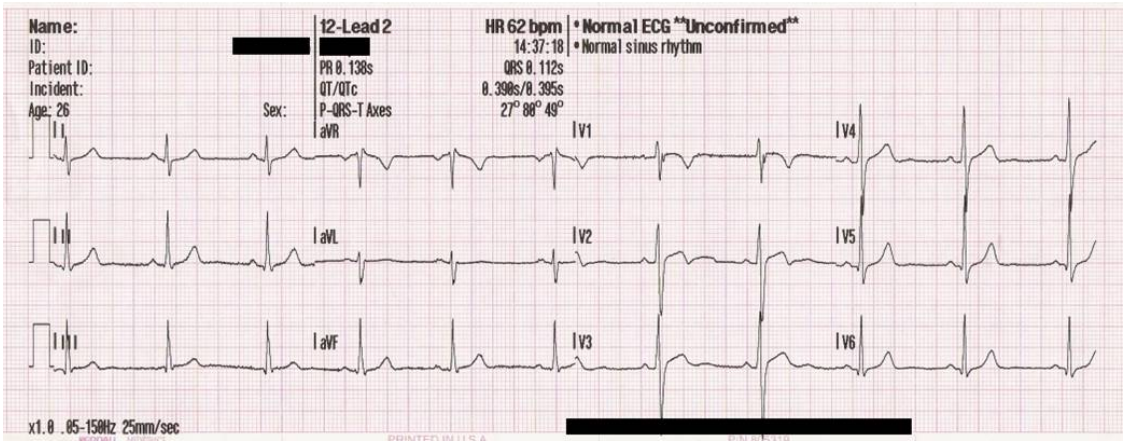
eventually result in heart failure. Rheumatic fever can occur at any age, but usually it happens in children ages and it is uncommon in the most developed countries [2, 8].

Disorders of the heart muscle, *i.e.* *cardiomyopathies*, and disorders of the electrical conduction system of the heart, *i.e.* *cardiac arrhythmias*, represent relevant CA even if they are less common than heart attacks and strokes, as shown in Figures 1.1(b) and 1.1(c). Cardiomyopathy is a wide term for CA related to the heart muscle where the walls of the heart chambers have become stretched, thickened or stiff [9]. As a direct result, the heart muscle weakens and becomes unable to pump blood to the rest of the body. Several kinds of cardiomyopathy are inherited and they can be noticed in children and younger people. Cardiac arrhythmias are a set of diseases in which the heartbeat rhythm is irregular, *i.e.* excessively fast, slow, or chaotic and unpredictable [10]. Usually, the normal heart beating rhythm in a resting subject is within 60 and 100 beats per minute (bpm) [11]. For instance, *sinus tachycardia* is a kind of arrhythmia that happens when the heart rate is superior to 100 bpm in adults. On the other hand, *bradycardia* happens when the heart rate is inferior to 60 bpm [10]. While several kinds of arrhythmia are not life-threatening, some of them may cause complications such as stroke or heart failure, and eventually sudden death. A well-known example is *ventricular fibrillation* in which heart ventricles suddenly stop pumping normally, due to sudden chaotic electrical activity. Ventricular fibrillation results in cardiac arrest with loss of consciousness and heart pulse, and it causes the majority of sudden deaths [12].

1.2 Automatic Interpretation of Cardiac Abnormalities

In Section 1.1 we pointed out that CA represent more than a half of the total number of deaths caused by CVD worldwide. Thus, physicians are making a big effort toward the early diagnosis and prompt treatment of CA [13, 14], which are of paramount importance for people who are at high cardiovascular risk [15]. Several invasive and non-invasive clinical tests are usually performed to diagnose the

Figure 1.2: An example of clinical 12-lead ECG recorded on an anonymized patient. Common clinical measures, *e.g.* bpm rate and degrees of cardiac axes, along with an automatic suggestion for the final diagnosis are provided by the ECG recording machine. Credits: MoodyGroove, Public domain, via Wikimedia Commons available at `https://commons.wikimedia.org/wiki/File:12leadECG.jpg`.



presence of CA, including blood tests, electrocardiogram, echocardiogram, coronary angiogram, and magnetic resonance imaging [15].

The Electrocardiogram (ECG) is considered as one of the most important clinical tools for the detection and diagnosis of CA [15, 16]. When performing an ECG, the heart electrical activity is recorded to assess weather the heart is beating under a normal condition. Several electrodes and wire leads are put on chest, arms, and legs. Then, the leads are connected to an ECG machine which is capable of recording the electrical impulses generated by the heart and printing them out on paper [17, 18]. Often, along with the recordings, the ECG machine provides common clinical measures, and automatic suggestions to help physicians in the final diagnosis of the possible CA. The standard clinical practice usually employs the 12-lead ECG, which is recorded through 10 electrodes (an example of a clinical 12-lead ECG is reported in Figure 1.2). One of the great advantages of the ECG is that it is widely available, giving physicians around the world an easy, rapid, non-invasive, reproducible, patient-friendly, and last, but not least, inexpensive way of obtaining a wealth of information about cardiac health [19].

Usually, a physician provides an *interpretation* of the ECG, where with the term "interpretation" here we specifically refer to the assessment process of the morphology

of the ECG waves and common time intervals to understand if one or multiple CA are present. However, the interpretation process is time-consuming and it requires a high degree of training [20]. It is then not surprising that the first attempts to build computer programs for the automatic interpretation of ECGs are dated back at the end of the 1950s, when it was soon expected that computers would have a crucial role in the process of ECG interpretation [21–23]. The traditional ECG interpretation programs were built *translating* into the machine the interpretation rules developed within the standard practice of physicians [21, 23], and their use has spread since the 1980s, when real-time analysis and direct print on paper of the results along with ECGs were introduced [24].

Despite the huge initial technical efforts, the clinical employment of computer programs to interpret ECG remained initially limited because of the lack of agreement on waves definitions, common measurements, and standardized criteria for interpretation [25]. To address such problems, efforts to propose standards and recommendations for the interpretation of ECG were developed worldwide to establish an international standard for the computerized interpretation of the ECG [26]. The aim was to reduce the wide variation in wave measurements and in the diagnostic interpretation of ECG, so that similar or at least comparable results could be obtained independently of the employed computer program [16, 25].

Even considering all the efforts and advances in the field of automatic interpretation of ECG, worldwide accepted standards for its interpretation are still missing [21]. However, yet in 1988, a survey report showed that over 50% of the 100 million ECGs recorded in the United States were interpreted by computer programs [26]. In the next decade such number doubled and by 2006 it was reported that 100 million ECGs were being interpreted by computers annually in the United States, and a similar number in Europe [26]. Nowadays, all the modern clinical ECG acquisition devices are equipped with automatic interpretation programs, which offer diagnostic proposals to assist the physicians' decision-making process while improving their diagnostic accuracy [27], and reducing the required time to get ECG interpretations [26]. For instance, in Hongo *et al.* it was estimated that

computer-assisted ECG interpretation decreased the required interpretation time by up to 24% to 28% for experienced physicians [26].

An example of ECG interpretation computer program is represented by the University of Glasgow (Uni-G) ECG analysis program that has been in continuous development for over 30 years [28]. Several other examples of automatic interpretation programs nowadays employed in the standard clinical practice are reported in De Bie *et al.* [24]. As a final remark, we notice the improvements lead by such ECG interpretation programs have shifted their role from saving the time of cardiologists, and improving their diagnostic accuracy, to even supporting the diagnostic process when access to a specialist is not possible. Such possibility was recently made available by the latest advancements of telemedicine, which involves the employment of reliable communication systems for remotely delivering biomedical signals over long distances to physicians, and to return back the diagnosis to the patients [29, 30].

It is clear that computer programs for ECG interpretation has had a huge impact on electrocardiography by assuming a large role in the diagnostic interpretation, where their most important advantages include the improved percentage of correct interpretation of ECGs [27], and the reduction of physicians reading time [26]. The computerized interpretation of ECG has evolved into a necessary tool for the modern medical practice, but the preliminary diagnostic interpretations offered by automated computer programs still come with several drawbacks [16].

The automated interpretations of ECGs are often wrong, where the most common errors involve for instance the interpretation of atrial fibrillation, pacemaker rhythms, and myocardial infarction [31, 32]. Shah *et al.* [33] assessed the interpretation performance of the General Electric GE-Marquette analysis program, which is a top-level ECG interpretation computer program provided by the GE Healthcare company (Milwaukee, WI, USA) [34]. The authors compared automatic ECG interpretations to that of two expert over-readers in assessing 2112 randomly selected standard 12-lead ECGs. The normal sinus rhythm[3] was correctly interpreted by computer programs in 95% of the ECGs with this rhythm. However, non-sinus

[3]The terminology "normal sinus rhythm" is often employed to denote a specific kind of rhythm where all common clinical measurements of the ECG fall within designated normal limits [18].

rhythms were correctly interpreted with an accuracy of only 54%. The automatic computer program interpreted sinus rhythm with a sensitivity of 95%, specificity of 66%, and positive predictive value of 93%. However, the automatic program interpreted non-sinus rhythms with a sensitivity of 72%, a specificity of 93%, and a positive predictive value of 59%. Several other recent studies which report related findings are presented in Estes [31] and De Bie *et al.* [24]. Thus, physicians must be aware of the hazards of relying on preliminary diagnostic interpretations, and automated interpretation of ECG must be often over-read by trained physicians to offer accurate diagnoses of CA. As a direct result, the automatic interpretation of ECGs must be be regarded as supplement, but not as a substitute of the interpretation provided by expert physicians [31].

Besides the potential interpretation errors, the accuracy of interpretation programs may even significantly vary according to both the manufacturer's program and the level of the ECGs over-readers [21]. Computer programs are usually tested in comparison with interpretations provided by several expert physicians, considered to be the *gold standard*. Indeed, the quality of interpretations provided by computer programs has been deeply questioned [24, 26]. Further, the ECG datasets employed for testing computer programs often poorly represent the overall population with respect to age, gender, and possible clinical diagnoses usually faced in the daily medical practice [21, 26]. Advanced comparative assessments of the accuracy of commercially available computer interpretation programs were rarely performed, mainly due to the reluctance of the manufacturers [26]. To the best of our knowledge, only recently a study from De Bie *et al.* [24] compared the most currently employed ECG interpretation programs on a wide dataset by assessing their accuracy in detecting CA, including for instance atrial fibrillation and flutter. The study not only confirmed that automatic interpretations could be often wrong, but they can even significantly differ between the analyzed computer programs. Thus healthcare institutions and physicians should not rely only on a selected interpretation program to decide the treatment of CA.

As a remarkable drawback, it must be noted that the majority of the commonly employed ECG interpretation programs come with proprietary licenses with high cost for hospitals and medical institutions that cannot meet the needs of remote areas for proper management of CA, in particular of low and middle-income countries [35]. Several less expensive novel ECG interpretation tools were developed, including for instance portable ECG monitors and wearable devices. Such tools come with different interfaces and functionalities that can potentially affect their accuracy, size, reliability, and power consumption. However, despite the huge efforts geared towards the development of less expensive ECG interpretation tools, their diagnostic accuracy, and reliability are still sacrificed, thus representing major issues [35].

Finally, as a consequence of their proprietary license, the computer programs for ECG interpretation are often partially or completely *opaque* to the final user, in the sense their source code is not accessible and they cannot be queried to understand the reasons behind the provided interpretations. Nevertheless, although not wanted by manufacturers, with the growth of automated ECG analysis the ECGs have become widely interpreted by less experienced physicians who nowadays often rely more and more heavily on opaque computer interpretation programs [24, 36]. Regarding this aspect, we still stress the fact that computerized interpretations come with several drawbacks, thus they should not be considered as a full replacement of the experienced cardiac physician [31].

1.3 Classification of Cardiac Abnormalities with Machine Learning

Machine Learning (ML) algorithms were applied in electrocardiography in the last decades to automatic interpret ECGs [37–41]. ML is a subfield of the well-known Artificial Intelligence (AI), where AI is a broader term which describes any computational program that *mimics* certain capabilities of the human intelligence, such as problem solving skills, by modeling them with explicit rules designed for the problem at hand [39, 42]. Even if AI and ML have been often used interchangeably in the context of electrocardiography, they represent different ways to automatically

address tasks with the use of computers [37, 39]. ML automatically learns how to address a specific task by discovering useful patterns from data, without using explicit instructions provided by domain experts [39, 43]. On the other hand, AI methods address tasks according to preset rules that are designed relying on the human knowledge [39, 43].

In the context of electrocardiography, researchers often modeled the task of ECG interpretation as a supervised classification ML problem [44, 45], *i.e.* they identified CA by training ML algorithms to learn a proper classification model from a set of labeled training ECG data, where the ground truth was provided by expert physicians [37, 39, 41]. In particular, the availability of public datasets, mainly shared by the Physionet project, allowed researchers to design and train promising algorithms to classify CA [46]. Several ML algorithms are available to learn a classification model from data with associated information on the outcome. The most typical supervised ML classification algorithms leveraged for ECG classification include support vector machine, k-Nearest Neighbors (k-NN), decision tree, random forest, and artificial neural networks [47–49].

The approach of the ECG interpretation computer programs introduced in Section 1.2 resembles to the one of AI and may result far from ML, which seek to exploit patterns within the available data to identify CA rather than relying on a fully empirical set of human-designed rules [23, 41, 43]. To stress the differences within the two approaches, let us consider a deeply simplified version of the ECG interpretation task where, for the sake of simplicity, we are required to interpret single-lead ECGs that show normal sinus rhythm or bradycardia[4]. The interpretation computer programs presented in Section 1.2, which rely on the clinical domain knowledge, usually take as input the ECG signal, apply a certain preprocessing to the signal, compute the bpm rate, and provide an interpretation relying on the 60 bpm threshold [23, 38, 41].

[4]Bradycardia is a condition wherein the resting heart rate is under 60 bpm in adults [10]. Nowadays, several models of smartwatch showed that it is possible to detect normal sinus rhythm and bradycardia from single-lead ECGs with remarkable performance [50].

An alternative to the usage of the domain knowledge of physiology in interpreting ECGs is represented by ML [37, 39–41]: the above mentioned simplified version of the ECG interpretation task can be modeled as a supervised classification ML problem, for instance by employing the logistic regression function to classify within the two considered CA. The logistic regression function is a common parametric ML classification model, borrowed from statistical learning [44, 45], which maps a real-valued input vector[5] $\boldsymbol{x}$, our sampled single-lead ECG, to a scalar prediction $\hat{y}$ in the range $[0, 1]$ as

$$\hat{y} = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b), \tag{1.1}$$

where $\boldsymbol{w}$ is a set of parameters, also called *weights*, b is an additive bias, and $\sigma(\cdot)$ is the logistic function with the following functional form:

$$\sigma(z) = \frac{1}{1 + e^{-z}}. \tag{1.2}$$

If we map the normal sinus rhythm to the value of 0 and bradycardia to the value of 1, the logistic regression function is a simple model which offers the computer a way to classify within the two considered CA. Since the logistic regression outputs a value between 0 and 1, we may think of it as a probability: a zero input (with $b = 0$) gives a value of 0.5, thus we could predict bradycardia whenever the probability is greater than 0.5, and normal sinus rhythm whenever the probability is less than 0.5. To properly calibrate the logistic regression model, a starting guess is set for the unknown parameters $\boldsymbol{w}$, usually by randomly sampling them [44, 45]. Then, the model is supplied with pairs of ECGs, usually called *features*, and corresponding diagnosis, usually referred as *labels* (in this case, zero or one values). A set of instructions is provided to fit the data to the underlying equation as an optimization problem by minimizing the prediction error, usually referred as *loss* or *cost function*. Finally the instruction set is continually executed to update the parameters to fit this data to the underlying equation with lower an lower

[5] If not differently stated, we hereinafter refer only to column vectors.

prediction error, and such is usually called as *training process*[6]. Even if simplistically represented, each emphasized term above identifies the basic building blocks of any parametric ML algorithm [44, 45], and if such blocks are tuned properly they allow for the development of novel techniques to classify ECGs [37, 39, 41]. Furthermore, we mention that even non-parametric ML algorithms exist, which act similarly to the parametric ones, but they do not involve the step of the calculation of the optimal parameters. A well-known example of non-parametric ML algorithm is represented by the k-NN algorithm where the class of a new sample is computed through the majority voting of the classes associated to the k samples with the closest distance to the new one (relying on some definition of distance) [44, 45].

Most of the supervised ML algorithms are not directly fed with raw ECGs, as we described in the above example. Instead, ML algorithms are usually fed with handcrafted feature vectors for the classification task to accomplish, computed relying on the domain knowledge of electrocardiography [37, 39–41]. For instance, peak amplitudes and time windows computed over most important ECG waves, frequency domain features, and statistical features computed on the ECG are among the most common ones [47, 48]. However, in the last decade we saw the development of a completely new approach called Deep Learning (DL), a research field belonging to the ML domain, where computers efficiently learn how to make automatic classifications in a fully data-driven way [51, 52]. With the advent of DL, the approach for tackling automatic classification problems moved from the calculation of handcrafted features to an innovative *end-to-end* learning strategy, where the classification model automatically learns the relevant features for the task to accomplish directly from the raw data (or slightly preprocessed).

The most common DL algorithms are represented by Deep Neural Networks (DNN) [44, 51, 52]. DNN are a kind of artificial neural network that consist of multiple simple non-linear models, usually called *neurons*, which compute a weighted sum of the inputs and threshold the resulting sum by setting it as input of non-linear functions. Neurons are usually arranged in series where each one is named *layer* [44,

[6]Stochastic gradient descent and binary cross-entropy are often employed in this case, respectively as training algorithm and loss function [44, 45].

45]. The more is the number of layers and non-linear neurons in each layer, the more the DNN are capable of catching complex information from data [51, 52]. By far, Convolutional Neural Networks (CNN) are within the most common kind of DNN used to classify ECGs[7], in which convolutional filters and subsampling operations are applied in cascade [38, 49, 54, 55]. The convolution operation considers a small pattern, usually referred as *convolutional kernel*, and it locates where such pattern arises in the input signal by multiplying the kernel itself with the input through a sliding window. The optimal pattern to search for properly classifying within the classes of the problem at hand is automatically learned from data during the training process [44, 51, 52]. Even by means of the subsampling layers, the convolution operation is capable of retaining useful information through successive layers by removing artifacts deemed unnecessary by the neural network during the training process [51, 52]. Usually, serial combinations in parallel and series of convolution and subsampling layers allow the CNN to learn simple concepts at each layer, that finally build up to learn more and more complex concepts. For instance, in the most intuitive example of CNN used in image recognition tasks, convolutional layers are capable of learning simple entities in the first layers, *e.g.* lines, circles, that finally build up into more sophisticated representations, *e.g.* beaks, feathers, eyes [51].

DL has seen a dramatic rise in the past decade due to the availability of large databases and new high-performance computing methods [51, 52]. Groundbreaking performance were delivered by DL in several research fields, such as speech recognition, image classification, and language translation, some of them at human-level performance [51]. For instance, in the context of computer vision CNN often showed recognition accuracy better than, or at least comparable, to humans in several visual recognition tasks [56], including recognizing traffic signs [57], faces [58], hand-written digits [59], facial landmarks [60], and on more generic image recognition tasks. For instance, He *et al.* [61] surpassed the human-level recognition performance reported by Russakovsky *et al.* [62] on a more generic and challenging recognition task involving the classification of images within $1,000$ different classes. The large

[7]Nevertheless, it must be noticed that a wide range of DL models do exist [53], even if they were less applied on the ECG classification task [48, 49, 54].

impact that DL had in several research fields has motivated the investigation of such methodologies for the automatic classification of ECGs [38, 49, 54, 55]. Indeed, private institutions have recently begun in collecting massive ECG databases that are orders of magnitude larger than the public ones previously proposed, and then they trained DNN (especially CNN) onto them [49, 54, 55].

Zheng *et al.* [63] collected a publicly available database consisting of $10,646$ ECGs, including $5,956$ males and $4,690$ females. Among those patients, 17% had normal sinus rhythm and 83% had at least one CA, including for instance sinus bradycardia, atrial fibrillation, and supraventricular tachycardia. Wagner *et al.* [64] proposed the PTB-XL[8] dataset composed of $21,837$ clinical 12-lead ECGs from $18,885$ patients of $10s$ length. The ECGs are publicly available and they were labeled with 19 classes, including for instance bundle branch blocks, myocardial infarction, and atrio-ventricular blocks. Hannun *et al.* [67] collected a private database of single-lead ECGs consisting of $91,232$ ECG records from $53,549$ patients which showed normal sinus rhythm and other 11 CA, including for instance atrial fibrillation, atrial flutter, and ventricular tachycardia. Recently, the PhysioNet project proposed two challenges within the Computing in Cardiology conference which asked participants to classify CA from 12-lead ECGs in 2020 [68], and from varying set of leads including 12-lead, 6-lead, 4-lead, 3-lead, and 2-lead ECGs in 2021 [69]. In both the editions, the challenge data was composed of annotated ECGs from six sources in four countries and across three continents. The second year challenge database included over than $130,000$ 12-lead ECG recordings with more than $88,000$ ECGs shared publicly as training data, while the remaining ones were retained as validation and test data. The available ECGs were annotated with multiple labels at the same time considering 133 possible CA, including for instance the ones we previously mentioned for the previous introduced databases. Zhu *et al.* [70] collected a dataset composed of $180,112$ 12-lead ECGs from $70,692$ patients and they labeled signals with 20 possible CA. Similarly to the 2020 and

[8]The acronym "PTB" stands for the Physikalisch-Technische Bundesanstalt, the national metrology institute of Germany which made available the first version of the database in 1995 for research, algorithmic benchmarking, or teaching purposes [65, 66].

2021 PhysioNet challenge databases, a portion the ECGs were annotated with multiple labels at the same time. Finally, Ribeiro *et al.* [30, 71] collected a massive and private ECG database composed of $2,470,424$ ECGs, recorded on $1,773,689$ patients. The database was collected starting from 2010 by the Telehealth Network of the state of Minas Gerais, Brazil [72], and ECGs were labeled with several common CA including atrio-ventricular block of first kind, bundle branch blocks, sinus tachycardia, bradycardia, and atrial fibrillation.

DNN may learn the optimal features for a specific classification task, thus likely outperforming ML algorithms that are fed with handcrafted ones[9], indeed they delivered promising performance on the above presented datasets [48, 49, 54]. For instance, Hannun *et al.* [67] implemented a CNN with residual connections inspired from He *et al.* [73], and the authors assessed the model's performance by asking to a board of expert cardiologists to manually annotate 328 test ECGs. The expert physicians performed worse if compared to the CNN in detecting all the 12 considered CA, except junctional rhythm and ventricular tachycardia. Ribeiro *et al.* [30, 71] implemented a residual CNN comparable to the one of Hannun *et al.* [67], but with fewer layers, with the same aim of classifying CA, but among 6 classes. At a larger scale with respect to Hannun *et al.* [67], the authors trained a CNN with residual connections in an end-to-end fashion to diagnose various CA on one of the largest available ECG databases (to the best of our knowledge), that we introduced in the previous paragraph. Similarly to the case of Hannun *et al.* [67], the classification performance of the trained model, assessed by its positive predictive value, sensitivity, specificity, and area under the receiver operating characteristic curve, was slightly better if compared with a cohort of medical trainees (thus, including even students). Finally, Zhu *et al.* [70] proposed a comparable study to Hannun *et al.* [67] and Ribeiro *et al.* [30, 71] by relying on a CNN with residual connections, but the authors widened the CNN approach to classify 20 CA from 12-lead ECGs with multiple labels at the same time. The proposed CNN was

[9]Even if it happened in other research fields, such as Computer Vision [56], it is worth noting that, to the best of our knowledge, the superiority of DL with respect to ML algorithms has not been proved yet on the ECG classification task.

validated on an independent test dataset composed of 828 patients' ECGs that had been annotated by a panel of three cardiologists, and the 24% of the ECGs in such dataset was labeled with more than one abnormality. The CNN correctly classified all the CA in 80% of the ECGs available in the test dataset, which it was also interpreted by 53 physicians, divided in three groups based on their experience. The average accuracy showed by the groups of physicians was 70%, thus worse than the one showed by the CNN, and even the physicians with more than 12 years of experience of ECG interpretation were less accurate than the CNN (they correctly interpreted 75% of the test ECGs).

1.4 Drawbacks of Machine Learning: Motivation of the Thesis

The main advantage of DNN is represented by the optimal feature representation achieved after the training phase [51, 52]. Their capability of automatically learning relevant features is due to the large amount of parameters that these models contain, which usually is in the order of *tens of millions* [74]. However, with such a large amount of parameters, the classification outputs provided by DNN become difficult to explain (or even impossible) [75–77]. These models are composed of multiple layers with several interconnected neurons, and each neuron is associated to different weights, with the aim of automatically extracting the relationship between the input and the respective output. With a huge number of weights it is practically unfeasible to understand how the neurons interact to determine why a certain output was provided. Thus, the price of this *luxury* in capturing complex data representations, which often lead to remarkable classification performance, is the aforementioned loss of model understandability which smears the reputation of DNN as *black-boxes* [75–77]. The perception of dealing with opaque models is mostly associated with the final users of DNN: even if computer scientists and engineers could potentially understand the architecture of DNN[10], the process by

[10] Always by considering that the state-of-the-art DL architectures are often composed of tens of millions of weights and non-linear operations [74], thus making it difficult to analyze their interior data flow.

which such models perform the classification can be inscrutable to humans, limiting the trust in them, and thus hindering their acceptance [75, 78, 79]. Finally, it must be noted that the above-mentioned concerns about missed understandability even hold for the majority of other ML models, which include the DL ones. The number of involved parameters, and the underlying complex architecture of most ML models makes it difficult to understand the reasons behind their classifications, likely to what happens with the DL ones [75–77].

In order to *open* ML black-boxes and understand why they provide their classification outputs, researchers introduced several approaches to explain models' outcomes, thus creating a new line of scientific research usually called "eXplainable AI" (XAI) [75–77]. Methods for reaching explainability were mostly developed in the computer vision domain, where researchers wanted to understand which portions of the input image were the most relevant to DL models for getting classification outputs [75, 77]. Another term that is frequently used to frame the XAI research is "interpretable ML", but it is worth mentioning that there is still no agreement within the ML community on the definition of the terms "interpretability" and "explainability" [75, 79]. Even if several authors attempted to distinguish between them, most use such two terms interchangeably. Thus, although a clear universal definition is still not available, for the sake of convenience in the following we will refer to the term "explainability" of ML algorithms when we refer to the development of techniques to explain the rationale behind the classification outputs of black-box models. In such way we conveniently avoid the possible confusion that could arise from the usage of terms linked to "interpretation", employed in the previous Sections 1.1-1.3 to consider the well defined task of ECG interpretation.

The scientific research of XAI is nowadays perceived as required since black-box models are currently being employed for taking high-stakes decisions throughout society, potentially causing critical problems in healthcare, criminal justice, and other domains [75, 79]. ML is currently leveraged for high-stakes applications that deeply impact human lives, and several of them are black-boxes that do not explain their outputs in a way that humans can understand. The lack of transparency

and accountability of predictive models can have (and has already had) severe life-threatening and ethical consequences [75, 79]; and generally poor use of limited valuable resources in criminal justice, medicine, energy reliability, finance, and in other domains [80]. Concrete examples where black-boxes negatively impacted human lives, even leading to life-threatening consequences, are reported in Guidotti *et al.* [75] and in Rudin [79]. A well-known case in which a black-box ML model was fatal for a person is represented by the case of the death of Elaine Herzberg [81]. Elaine was the first recorded case of a pedestrian fatality involving a self-driving car, after a collision that occurred in 2018. She was pushing a bicycle across a four-lane road in Tempe, AZ, USA, when she was fatally struck by an Uber test vehicle, which was operating in self-drive mode with a human safety backup driver sitting in the driving seat. The full reasons of why the autonomous car did not stop are still not completely understood, but Michael Ramsey, a self-driving car expert with the Gartner company (Stamford, CT, USA), characterized the fatality episode as "a complete failure of the system to recognize an obviously seen person who is visible for quite some distance in the frame." [82].

Regarding the context of electrocardiography, and even the broader context of healthcare, ML is more and more being conceived as a technology with a real potential to transform such fields, by allowing to be leveraged for high-stakes applications [23, 83–85]. ML could potentially enhance the decision-making capabilities of the individual clinicians by improving the accuracy of their diagnoses, and by reducing the required time for obtaining them, thus to promptly apply proper treatments. At the institutional level of healthcare, ML could improve its inefficiencies in the workflow, potential waste of resources, inequities, and exploding costs [83–85]. However, even if we accept the premises of this *exciting* narrative, the enhancement of clinicians and healthcare institutions by means of ML is less straightforward than it might appear. The employment of ML in healthcare goes hand in hand with several trade-offs on the epistemic and ethical level. Even if there is plenty of evidence of ML algorithms *outsmarting* their human counterparts[11], their

[11] Examples in the context of electrocardiography are represented by several works we introduced in Section 1.3, *e.g.* Hannun *et al.* [67], Ribeiro *et al.* [30, 71], and Zhu *et al.* [70].

deployment comes at the costs of high degrees of uncertainty. Even if employing ML might improve the accuracy of medical diagnosis and the time required to obtain it, it comes at the expense of opacity when assessing the reliability of the provided diagnosis itself, thus leading to an impoverished outcome. From an ethical point of view, deferring to ML *blurs* the attribution of accountability, and thus exposes patients to severe health risks [83].

In our view, none of the presented concern presents a definitive *knockout* argument against the employment of ML within ECG classification, and even healthcare. On the contrary, we are convinced that ML provides several opportunities to enhance the reliability and the time required for decision-making in such contexts, if ML models could be turned into *glass-boxes*. In this respect, the involvement of ML in healthcare decision-making might yield more valuable outcomes, along with a better epistemic, and consequent ethical, reflection. However, even if for several research fields the desire of a deep understanding of ML applications is crucial and obvious [76], the demand of explainability is not perceived as always required in the context of healthcare [78, 83, 86]. Physicians cannot always provide a complete explanation of why they arrived at a particular diagnosis. Several effective drugs including aspirin, acetaminophen, and penicillin, were in widespread use for decades before their mechanism of action was understood [86], thus opaque decisions are more common in medicine than the most realize [78]. Indeed, even if our knowledge about the mechanisms of the human body is not completely known, physicians will not stop in treating people: even Aristotle noted over two millennia ago that when our knowledge of a causal systems is incomplete — as it often is in the medical practice — the ability to explain how results are obtained could be less relevant than the ability to produce such results and empirically verify their accuracy [87, 88].

Thus, should this thesis work raise ML to higher explanatory standards than physicians? An effort in the direction of developing explainability methodologies relies on the fact that we believe necessary that future advancement in automatic classification of ECG progresses together with our capability of understanding the classification outputs of ML models. Even if we pointed out that opaque decisions

are frequent in the context of medicine, this is not the case of ECG interpretation. The physical phenomena that generate the electrical activity which make the heart continuously beat have been well established during the last century [89], and the same holds for the standard way of measure it from the body surface using the 12-lead ECG, that the American Heart Association standardized at the beginning of the 1950s [14]. Finally, well established guidelines to interpret ECGs are available [17, 19], and physicians are nowadays pursuing more and more the challenge for their further standardization [16]. As a consequence, we think that researchers should investigate how to *demystify* black-boxes to properly employ them for high-stakes decisions in the context of electrocardiography. The rationale behind their classification outputs must be questioned, similarly to what happens with physicians which are able to provide the reasons behind the interpretation of certain CA. Only in this manner we can attempt to foster the trust in ML and its acceptance in the community of electrocardiography, by even facing the related ethical issues [54, 86].

In case a ML model shows high classification performance, we may expect that it classifies ECGs by exploiting patterns which are meaningful in the underlying domain, or at least correlated with the typical ECG markers that physicians usually assess on ECGs to provide their interpretations. However, there is no guarantee that ML models learn such meaningful patterns to consequently achieve the desired classification output [90]. Indeed, ML is only highly capable of catching regularities from data to optimally perform its classifications, regardless of the knowledge domain. As a consequence, ML decision-makers cannot be trusted only relying on their predictive performance which are evaluated on the available dataset. Moved by analogous motivations to the ones we reported, several researchers investigated explainability within ECG classification, and they assessed the characteristics of the ECG that were significant in the final classification output of ML models [54]. For instance, Strodthoff *et al.* [91] and Baalman *et al.* [92] highlighted the samples belonging to a single ECG beat that mostly contributed to the final classification output, respectively for CNN and DNN models. Mousavi *et al.* [93] showed how to highlight which ECG waves, beats, or combination of beats were important for the

classification output provided by deep recurrent neural network. Finally, Zhang *et al.* [94] highlighted the contribution of each ECG lead for the final classification output of a CNN with residual connections.

Despite several works addressed the problem of explainability in ECG classification, they are often focused on DL models, thus considering less other kind of ML algorithms. Furthermore, most of the introduced explainability approaches provide explanations limited to a few test ECGs, and they do not provide explanations framed into the domain knowledge of electrocardiography, *i.e.* by comparing them to the standard physiological guidelines for diagnosing CA. For instance, Strodthoff *et al.* [91] and Baalman *et al.* [92] limited to provide the time samples belonging to a single ECG beat that mostly contributed to the final classification output, without framing such explanations into the physicians' knowledge. On the other hand, Mousavi *et al.* [93] framed the explanations into the knowledge domain by pointing out which ECG waves, beats, or combination of beats were important for the final classification output. However the computed explanations were still limited to a single ECG at a time, thus lacking the possibility of systematically evaluating the average performance of a DL model against the domain knowledge, over the entire training dataset. Zhang *et al.* [94] provided interesting lead-level explanations, but such explanations are not linked to the physicians' domain knowledge for most of the CA considered in the mentioned article. For instance, knowing that an ECG was classified as atrial fibrillation because the underlying CNN relied more on certain leads than on others would not be so relevant in the standard clinical practice: a chaotic and unpredictable rhythm, typical of atrial fibrillation, may be observed on any of the 12 ECG leads [50, 95]. Finally, it must be noticed that all the mentioned articles explained DL classifications relying on a reduced set of CA. Baalman *et al.* [92] and Mousavi *et al.* [93] classified within normal sinus rhythm and atrial fibrillation, Strodthoff *et al.* [91] classified within normal sinus rhythm and myocardial infarction, and Zhang *et al.* [94] classified within normal sinus rhythm and other eight CA.

Even if a few methods have been introduced to gain more insight into the parameters learned by DL models, another major issue is represented by the underfitting and overfitting phenomena [44, 45]. Underfitting happens in the case a trained ML model obtains poor classification performance both on training data and on unseen data. On the other hand, overfitting happens when the ML model shows good performance on the training data, while poor performance on unseen data. To prevent such issues, the proper ML/DL pipeline composed of preprocessing, feature engineering (only in case of ML algorithms), and classification algorithm, along with its parameters and hyperparameters, must be tuned relying on the experience of the computer scientist or engineer [45]. Bad fitting could potentially arise in several frequent situations [44, 45]. For instance, underfitting may be observed when a ML model with too few parameters is not capable of catching patterns in data showing high complexity [44, 45]. On the other hand, overfitting may happen when ML models with a huge number of parameters are trained on datasets with limited size, thus failing in learning general patterns to classify data [96, 97]. Further, the overfitting phenomenon may also happen in the presence of a wide range of biases potentially hidden in the dataset [51, 97]. Regarding the latter concern, for instance in one of our previous studies (which we will present in Section 4.2) we found that even with a standard ML model, a Random Forest (RF), it was possible to achieve high classification accuracy for the automatic classification of myocardial infarction, even though we discovered that the RF was not relying its classification outputs on the ECG segments reported in the international guidelines for ECG interpretation [98]. Such behavior was probably caused by a bias present in the available dataset in which the ECGs associated with myocardial infarction were sampled from an elder population, with respect to the younger one from which normal sinus rhythm was sampled. Similar results come from a few other recent studies that we introduced in the previous paragraph [91–94].

To avoid the arising of bad fitting models it is thus essential to consider the quality of the dataset, which, if poor enough, may never be overcompensated by any degree of ML models adjustments [44, 97]. Within the ECG classification task,

with poor quality we refer to: 1) The classes imbalance that is usually present in the latest presented datasets [68, 69], and in the previous ones introduced by the PhysioNet project [46], due to several limitations during acquisition time (*e.g.* difficulty in collecting rare CA, or unavailability of patients showing the desired CA) [99]; 2) The limited acquisition of such datasets from a single site or relying on a single device manufacturer [68, 69]; 3) The limited significance of such datasets in representing the overall population with respect to age, gender, and ethnicities [100]; 4) Last, but not least, the CA associated to the ECGs shared with the latest presented datasets are usually validated by expert physicians, but in certain cases there is still no objective gold standard for ECG interpretation [100]. Further, the level of experience of the involved physicians has often being questioned and, to the best of our knowledge, the problem of the assessment of the expertise level of employed human annotators has not been deeply faced yet. However, it must be noted that, if not tackled, it might expose us to a concrete risk of overoptimistic rating ML and DL accuracy due to a low expertise of the human ECG readers, as even pointed out by Sinnecker [100].

Several of the state-of-the-art datasets we introduced in the previous paragraphs come with some of the presented issues. In Zhu *et al.* [70], the number of ECGs within different classes of the dataset significantly differs of orders of magnitude. The ECGs were collected relying only on GE-Marquette ECG machines (manufactured by GE Healthcare, Milwaukee, WI, USA), and on a Holter machine manufactured by the DMS Holter Company (Stateline, NV, USA). Further, the collected ECGs were only recorded in Wuhan (China), which makes it difficult to predict the accuracy of the network in interpreting ECGs from patients of different ethnicities [100]. Finally, it is difficult to assess the level of experience of the 53 physicians who labeled the ECGs in the test dataset, and on which the authors relied for the performance assessment of the introduced model. Similar concerns hold for the work of Hannun *et al.* [67] where the employed dataset was not class-balanced. ECGs were recorded by uniquely relying on the Zio monitor, which is a Food and Drug Administration (FDA)-cleared, single-lead, and patch-based ambulatory ECG monitor [101]. No

details were provided on the ethnicity of the involved patients, however their average age and sex were limited to 69 ± 16 years on the training set, 43% women, and 70 ± 17 years on the test dataset, 38% women. Finally, nine expert physicians were split into three panels, and each panel annotated one-third of the test dataset to generate the *gold standard*. It is difficult to assess if each of the panels was composed of physicians with comparable experience, thus if the entire dataset was labeled with homogenous fidelity. In Ribeiro *et al.* [71], the used dataset is unbalanced even if it is within the largest available ones to the best of our knowledge. The acquisition was limited to 811 counties in the state of Minas Gerais, Brazil and ECGs were recorded relying only on two tele-electrocardiograph devices manufactured in Brazil. Further, CA labels were not only validated relying on expert physicians, but even relying on a cohort of medical trainees (thus, including even students).

We must notice that the limitations related to having at disposal a single acquisition site, a limited kind of acquisition device, a restricted set of patients that could be not fully representative of the entire population (in terms of age, gender, ethnicity, *etc.*), and difficulties in the assessment of the experience of employed physicians are challenging problems that are not straightforward to tackle, due to potential limited funding and/or strict government regulations. However, in the context of ECG classification, researchers tried to mitigate the mentioned drawbacks by artificially altering the employed dataset or by modifying the architecture of the employed ML model. For instance, researchers improved classification performance by applying standard oversampling, which randomly duplicates samples of the minority classes [67, 102]. On the other hand, several articles made use of the undersampling technique, to undersample the over represented classes, where one of them is usually represented by the normal sinus rhythm [103]. Finally, researchers attempted to tackle the problem of class imbalance by modifying the loss of employed DL models [104, 105].

Despite the efforts made to address the classes imbalance problem, to the best of our knowledge there is neither a universal agreement nor a thorough study on the most effective algorithms to be employed to address this issue in the task

of ECG classification, and only preliminary comparisons of those methods are available for restricted sets of CA [102]. Further, even if several techniques are available to address the problem of data imbalance [99, 106], most of them were poorly explored probably due to their difficult fitting to the ECG classification task. For instance, an interesting method is represented by cost-sensitive learning, which assigns different cost to misclassification of samples from different classes [107], and it can be implemented in various ways depending on the underlying ML or DL algorithm. For instance, a common way to implement it is to train a DNN to minimize a certain misclassification cost, instead of the common employed loss functions [107]. Even if cost-sensitive learning could significantly improve the classification performance, the application of this method is only feasible to the cases where misclassification costs are known [106]. Unfortunately, it is quite challenging and sometimes impossible to determine misclassification costs in certain domains, including the ECG classification [99]. Properly setting misclassification costs may be not straightforward since in most of the cases they are unknown, and/or they cannot be given by domain experts [108]. Finally, it must be noted than in certain cases no rebalancing technique could address the biases that are present in most of the available dataset. For instance, let us refer again to our previous mentioned work where we noticed that a RF was capable of achieving high classification accuracy for the automatic classification of myocardial infarction, even though it was not relying on ECG segments reported in the international guidelines for ECG interpretation [98]. We recall that such behavior was probably caused by a bias present in the available dataset in which the ECGs associated with myocardial infarction were sampled from an elder population, with respect to the younger one from which normal sinus rhythm was sampled. In this case no rebalancing technique could address the problem, since any of them would *carry around* the bias itself.

1.5 Contributions of the Thesis

ML and DL seem to represent promising tools for automated ECG classification on the analyzed datasets. However, the latest research works which leveraged

them carry several drawbacks that were presented in Section 1.4, and some of them were tackled throughout the doctoral experience. First, we discussed that to let ML algorithms to perform at their best, the proper ML pipeline, composed of preprocessing, feature engineering, and classification algorithm, along with its parameters and hyperparameters, must be selected. Even when end-to-end DL algorithms are adopted, and the feature engineering step is learned from data, the optimal model architecture is crucial to get the best performance, and it must be determined relying on the experience of the ML expert. Furthermore, most of the ECG datasets provided to train ML and DL models come with a limited number of unbalanced classes, which could potentially lead to bad fitting phenomena, including overfitting and underfitting.

To address the above-mentioned issues, in Bodini *et al.* [109] we designed an ensemble ML classification algorithm to classify 27 CA. We employed a dataset that was not only multi-label, but it even widened the number of classes with respect to the previous datasets we presented in Section 1.3. Differently from most of the previous studies which leveraged DL, that often simply borrowed DL architectures from other domains, each ML model in the ensemble was designed according to the domain knowledge of electrocardiography. In particular, each model classified a subset of the considered CA that alter the same set of ECG physiological features, known by physicians. Finally, the classification outputs of each model were concatenated to provide the requested output for the full set of CA. In Bodini *et al.* [110] we experimented three different Automated ML (AutoML) frameworks to automatically find the optimal ML pipeline in the case of standard and end-to-end DL algorithms to classify within 30 CA. The classes distribution of the used dataset was not balanced since several classes were provided with few training ECGs. To address this issue, cost-sensitive learning was leveraged: we run the AutoML frameworks to train the underlying ML and DL models by minimizing a custom misclassification score, instead of the commonly employed ML and DL loss functions. The misclassification cost was defined for each of the considered CA by expert physicians in the work of Perez Alday *et al.* [68].

Even if we saw in Section 1.3 that several research articles offer remarkable results for the value of ML and DL in classifying ECGs, we realized that only a handful of them offer insights into the models' learning representations of ECGs. Methods for opening black-boxes have been applied to the ECG in a few works and they were mostly limited to DL models. Furthermore, the presented methods provided explanations limited to a single ECG at time, they explained a few CA classes, and they came with limited framing in the knowledge domain of electrocardiography. To tackle the presented issues, in Bodini *et al.* [98, 111] we developed two frameworks to unveil which portions of the input ECGs were the most relevant to the classification output for both standard ML and end-to-end DL algorithms. With respect to most of the works that addressed the issue of explainability in ECG classification, presented in Section 1.4, we computed average explanations over all the training samples, and we translated them for the physicians' understanding. Furthermore, in Bodini *et al.* [111] we significantly widened the number of explained classes with respect of the analyzed works.

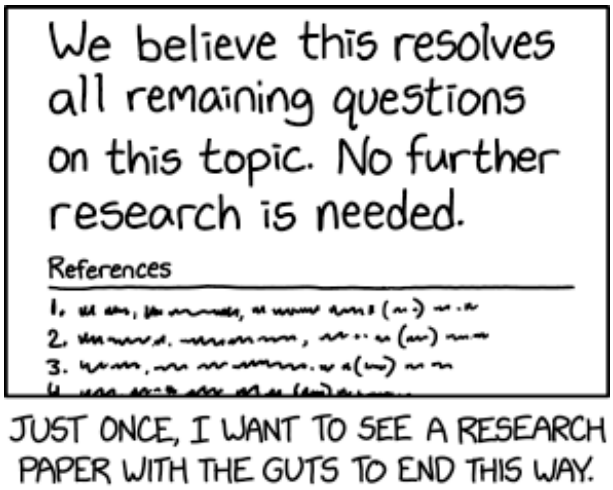
In particular, in Bodini *et al.* [98] we relied on the Local Interpretable Model-agnostic Explanations (LIME) algorithm to highlight which ECG leads were the most relevant for a RF algorithm in the classification of several kinds of ST-Elevation Myocardial Infarction (STEMI), depending on their anatomical localization. In the work we even showed how to overcome the overfitting problem caused by an inherent bias that was present in the dataset: we designed a ML classifier which relied on the domain knowledge of electrocardiography by setting as input of the RF model the proper ST-segment, which international guidelines for STEMI diagnosis suggest to focus on when interpreting it [112]. Furthermore, differently from the majority of other works presented in Section 1.4 which addressed the problem of explainability in ECG classification, we did not limit to provide the ECG time samples that were important for the classification of a single ECG at a time. We properly framed the explanations in the domain knowledge of electrocardiography by designing a custom metric which allowed us to highlight the importance of each lead in the final classification output, and we averaged the explanations over all the

training ECGs. Differently from the other works presented in 1.4, such as Zhang *et al.* [94], a meaningful framing for ML explanations was selected depending on the considered CA: a lead-level analysis was adopted in this case since it is suggested for the correct diagnosis of STEMI and its anatomical localization [112].

In Bodini *et al.* [111] we designed two explainability frameworks relying on two explainability algorithms for CNN to understand which ECG waves (the P wave, QRS complex, and T wave) were the most relevant in the classification of 27 CA for a state-of-the-art CNN. The introduced frameworks could be useful from the perspective of the ML expert, since they allow to inspect if the trained network correctly relies on the same ECG segments which physicians are expected to look at in the usual diagnosis of CA. If not, the ML expert can be aware of it and he can address the issue by guiding the architecture towards the domain knowledge of electrocardiography. From the perspective of physicians, the proposed frameworks allow them to understand whether the classification output provided by the network relied on their domain knowledge, by highlighting the expected ECG waves assessed during diagnosis, and thus fostering the trust in the employment of DL. To the best of our knowledge, with the mentioned work we were the first, at the same time with Zhang *et al.* [94], to systematically evaluate the performance of a CNN against the domain knowledge of ECG interpretation. Furthermore, with respect to the other introduced approaches in which were considered a limited amount of classes, the evaluation was performed on a wide set of 27 different CA.

The thesis work is composed of other three chapters that are organized as follows: in Chapter 2 we will provide a review of the state-of-the-art methods for ECG classification and ML explainability. Regarding methods for ECG classification, we will report the most common noise reduction techniques for ECGs, the common extracted features from ECGs, and a wider review, with respect to Section 1.3, containing further works which focused on CA classification from ECG and made use of ML and DL algorithms. Regarding methods for ML explainability, we will present a categorization useful both to conveniently introduce some of the most common explainability approaches, including the ones we employed in Bodini *et al.*

[98, 111], and to understand similarities and differences between them. In Chapter 3 we will introduce the contributions of the works of Bodini *et al.* [109, 110] which focused on the design of ML and DL algorithms for classification of CA from ECGs. In Chapter 4 we will show the contributions of the works of Bodini *et al.* [98, 111] which addressed the problem of explainability of ML and DL algorithms in the same context of CA classification from ECGs. After the mentioned chapters, we will finally report the employed bibliographic references and the list of personal publications referred to the context of the present thesis.



Credits: xkcd, Creative Commons Attribution-NonCommercial 2.5 License, available at `https://xkcd.com/2268`.

2

State of the Art on Electrocardiogram Classification and Machine Learning Explainability

Contents

2.1 State of the Art on Electrocardiogram Classification

In Section 1.1 we pointed out that CA are within the leading causes of mortality worldwide, and their early diagnosis and prompt treatment significantly contribute to preserve people health and life. The ECG records the electrical impulses generated by the heart, which may show regular or irregular beating activity. As we reported in Section 1.2, computer programs provide fast and accurate tools for identifying CA through the ECG analysis and they have achieved more and more great success in supporting physicians. Then, in Section 1.3 we described that several of the latest computational diagnostic techniques which analyze ECGs for estimating the presence of CA are based on ML and DL. In the present Section 2.1 we will focus on them, and we will present the standard procedures carried when applying such methods, which usually include ECG denoising and feature engineering. Finally, further relevant ML and DL classification algorithms along with research works that classified ECGs relying on them will be presented, in addition to the ones introduced in Section 1.3.

2.1.1 Denoising of the Electrocardiogram

Predominant Noises in the Electrocardiogram

Healthy ECGs are time-varying signals with associated low amplitudes in the range $10\mu V - 5mV$ [48]. Their usual value is around $1mV$, and their frequency bands are in the range $0.05 - 100Hz$ [113], where the majority of them are within the range $0.05 \sim 35Hz$ [48]. To obtain accurate and trustable classification outputs, the majority of ECG classification algorithms need relatively noise free ECGs [48, 114, 115]. Nevertheless, ECGs are frequently corrupted by noise signals and artifacts, such as baseline wander caused by patients movements, Power-Line Interference (PLI), Electromyographic (EMG) noise, and many others, that may cause the deformation of ECGs, thus affecting the final classification process [18, 114].

Baseline wander and unexpected drift noise signals are caused by patient movements, respiration, inadequate electrode positioning, and variations in their skin impedance [18, 114]. For instance, baseline wander is a considerable source

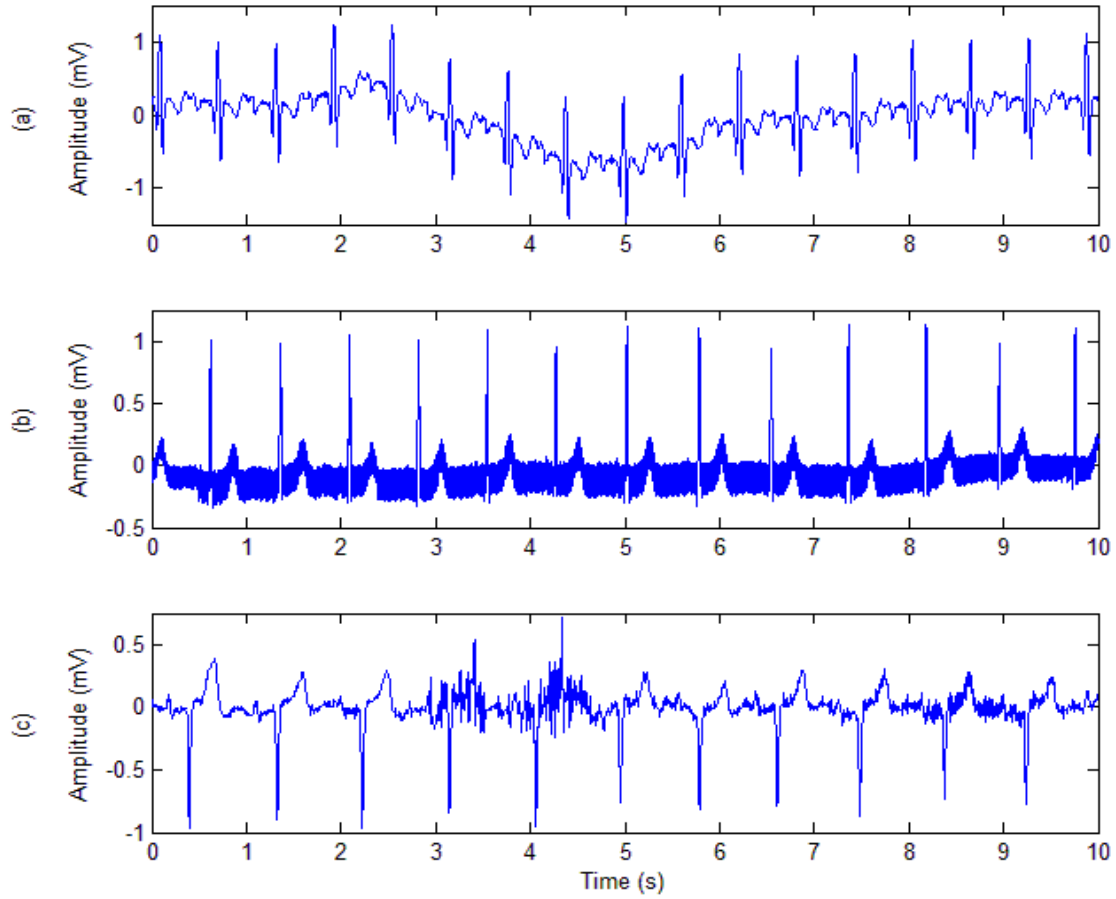


Figure 2.1: Common kinds of noise signals which corrupt ECGs. (a) Baseline wander, (b) PLI with $50Hz$ frequency, and (c) EMG noise. Credits: Maggio *et al.* [116], Creative Commons Attribution 3.0 Unported License, via IntechOpen available at the following url `https://www.intechopen.com/chapters/27012`.

of noise in several situations including Holter monitoring and ECG analysis in a moving ambulance, as well as during physical workout [48]. An example of ECG with baseline wander is reported in Figure 2.1(a). The amplitude of baseline wander noise ranges by $\sim 15\%$ of the peak-to-peak ECG amplitude [48] and its spectral content is usually confined to an interval well below $1Hz$, but it may contain higher frequencies during strenuous exercise [18]. Intense baseline wanders or movement artifacts may corrupt several low frequency elements of ECGs [48, 115]. For instance, baseline wander usually alters the ST-segment of the ECG, which may result in an incorrect classification of myocardial infarction and several other ST-segment associated CA [117, 118].

The PLI noise shows a bandwidth usually within $50/60 \pm 0.2Hz$, and an amplitude

of about the half of the peak-to-peak ECG signal amplitude [115]. The PLI noise is primarily caused by the pervasive electric lines in ECG machines' sampling circuits, by improper grounding of such machines, and interference caused from nearby equipment [18]. An example of ECG corrupted with PLI noise is reported in Figure 2.1(b). The PLI noise components overlap with the ECG frequency content, thus often distorting the morphological properties of its low-amplitude components [48, 115]. For instance, PLI noise may cause aberrations of the P wave in ECGs which lead to an incorrect diagnosis of several atrial-related CA, such as atrial enlargement and atrial fibrillation [115]. The PLI noise can be partially reduced by employing adaptive filters [115], by using proper electrical insulation, by preventing loose wire connections, and by properly placing electrodes [48].

The electrical activity generated during muscles contractions gives rise to the EMG noise [18, 48, 115]. The EMG noise has a frequency bandwidth that ranges within $0.01 - 100Hz$, and an amplitude of around 10% of the peak-to-peak ECG signal amplitude [115]. An example of ECG corrupted with EMG noise is reported in Figure 2.1(c). The EMG noise can either be intermittent, *e.g.* due to a sudden body movement, or have more stationary noise properties [18]. Past research works found that artifacts lead by the EMG noise substantially corrupt the morphology of most of the ECG waves because the EMG noise frequency band is significantly overlapped to the ECG one [48, 115]. Therefore, removing artifacts lead by the EMG noise without corrupting relevant clinical features of ECGs is often a challenging task [48, 115].

Methods for Denoising the Electrocardiogram

We reported that the frequency content of ECGs is within the $0.05 - 100Hz$ bandwidth [113]. Several noise signals whose frequency content falls within the same band may significantly corrupt the relevant features of ECGs useful to classify CA [48, 114, 115]. As a result, ECG classification algorithms strongly need the reduction of noise without missing the key clinical features of ECGs. Reducing the corruption caused both by high frequency and low frequency noise signals on the ECGs, and thus enhancing the signal-to-noise ratio, is a critical step in any

algorithm designed for automatic classification of CA [18, 48]. Indeed, an important reason behind the success of computerized ECG analysis was the capability of improving the signal quality of ECGs by leveraging signal processing algorithms to denoise them [18]. It is not surprising that most of the currently available ECG acquisition devices employ hardware filters to reduce the noise in sampled ECGs [48]. However, adjusting the filtering parameters is not always straightforward, and it may lead to poor noise reduction or even to deformation of signals if not properly done. Thus, ECG denoising algorithms have been more and more widely developed to effectively reduce undesired noises from ECGs [115].

Digital filters are broadly employed to remove unwanted signals in the ECGs [48, 114], and they are uniquely identified by the discrete-time Fourier transform of the time response in the frequency domain [119]. Digital filters are classified as Finite-duration Impulse Response (FIR) filters and Infinite-duration Impulse Response (IIR) filters. If compared to FIR filters, the coefficients of IIR filters are set using a feedback difference equation. If the appropriate coefficients are selected, a wide family of low-pass, pass-band, and high-pass digital filters can reduce several noise signals. However, because of the wide frequency band and different amplitudes of corrupted ECGs, the noise removal effect of filters with fixed cutoff frequencies resulted often limited [48, 115].

The Wavelet Transform (WT) is commonly employed in the process of ECG noise reduction because of their remarkable time frequency properties [115, 120]. WT addresses a major disadvantage of the Fourier Transform, which is only capable of capturing global frequency information [119]. On the other hand, WT decomposes a function into a set of wavelets, *i.e* wave-like oscillations that are localized in time, along with a scale. WT convolves a signal with a set of wavelets at a variety of scales to compute their presence at a particular scale and location. In such way WT can locally extract spectral and temporal information at the same time [119]. Thus, WT is capable of decomposing signals into high frequency detail coefficients and approximation low frequency coefficients [119]. Since noise components are commonly present in the detail coefficients, the effects of noise corruption can

be reduced by applying threshold quantization on the detail coefficients. Then, ECGs can be recovered via wavelet reconstruction of the low frequency and high frequency coefficients [48, 119].

Selecting the proper threshold functions is essential for obtaining the required noise filtering effect when leveraging WT [119]. In particular, the effectiveness of a threshold strategy is mainly determined by the kind of threshold method and threshold criteria designed for the considered scenario [121]. Hard and soft threshold functions are commonly employed for ECG noise reduction [115]. The recovered ECGs with hard threshold approaches usually show better approximation properties, but such approach may cause the reconstructed ECGs to visibly oscillate, whereas the recovered ECGs with soft threshold methods usually show better smoothness, while coming with higher reconstruction error. To address this issue, several effective techniques were introduced to reduce the noise from ECGs relying on stochastic parameters adjustment [115]. For instance, the β-hill climbing approach is an optimization method able to generate search trajectories in an hypothesis space until a local optima is reached [122]. Alyasseri *et al.* [123] coupled the $\beta-$hill climbing algorithm with the WT to denoise ECGs, by exploiting it to obtain the optimal wavelet parameters that resulted in the smallest mean square error between the original and the resulting denoised ECGs. The introduced technique performed well in particular for ECGs corrupted with low frequency noise, and it was able to provide denoised ECGs with overall good quality.

To exploit the advantages of both hard and soft thresholds, Han *et al.* [124] introduced an enhanced wavelet denoising method, namely the Sigmoid function-based thresholding method, that is a compromise between the two approaches. To some extent, the introduced threshold method perform well in preserving the amplitudes of the principal distinctive ECG peaks. Üstündağ *et al.* [125] introduced a technique for denoising ECGs using a fuzzy based threshold scheme and wavelet analysis. The authors employed a loop-based approach to set the optimal parameters of the fuzzy membership function, and they identified the right threshold and variance parameters for obtaining optimal denoising performance. If compared

to soft and hard based thresholding strategies, the proposed one was capable of outperforming both of them by showing better denoising performance.

The Discrete Wavelet Transform (DWT) offers remarkable noise reduction performance for high frequency noise signals, but it often leads to the loss of crucial information at low frequencies [48, 119]. Singh *et al.* [126] designed an ECG denoising strategy based on DWT and non-local mean (NLM) estimation. ECGs corrupted by noise were decomposed into low and high frequency detail and approximation coefficients by using a two-level DWT decomposition. A threshold was applied to the two-level detail coefficients to reduce the high frequency noise. Because the second-level low frequency coefficients included the majority of the ECGs, the NLM estimation for the second-level approximation coefficients was computed independently to reduce low frequency noise. The authors showed that the introduced approach could reduce noise in low-frequencies more effectively and faster on the MIT-BIH arrhythmia dataset [127], with respect to the previous approaches.

A wide range of further filtering methods is available in the literature which allow to reduce the noise in ECGs, such as the Empirical Mode Decomposition method, which is a common alternative to the wavelet analysis [128]. To delve further into the literature related to ECG noise filtering, a complete reference is provided by Chatterjee *et al.* [115].

2.1.2 Clinical Perspective of Electrocardiogram Features

Since the ECG directly reports the electrical impulses induced by cardiac muscles, it displays the regular (or potentially irregular) beating function of the heart. Thus, it is critical to retrieve as much clinically relevant knowledge as possible from ECGs after the proper noise reduction step, introduced in Subsection 2.1.1. The ECGs are composed of a large number of time samples on which it can be computed a wide range of features that reflect their characteristics. Amplitude and durations computed over the P wave, QRS complex, and T wave are within the most considered features in the context of automatic ECG classification [47, 48, 114].

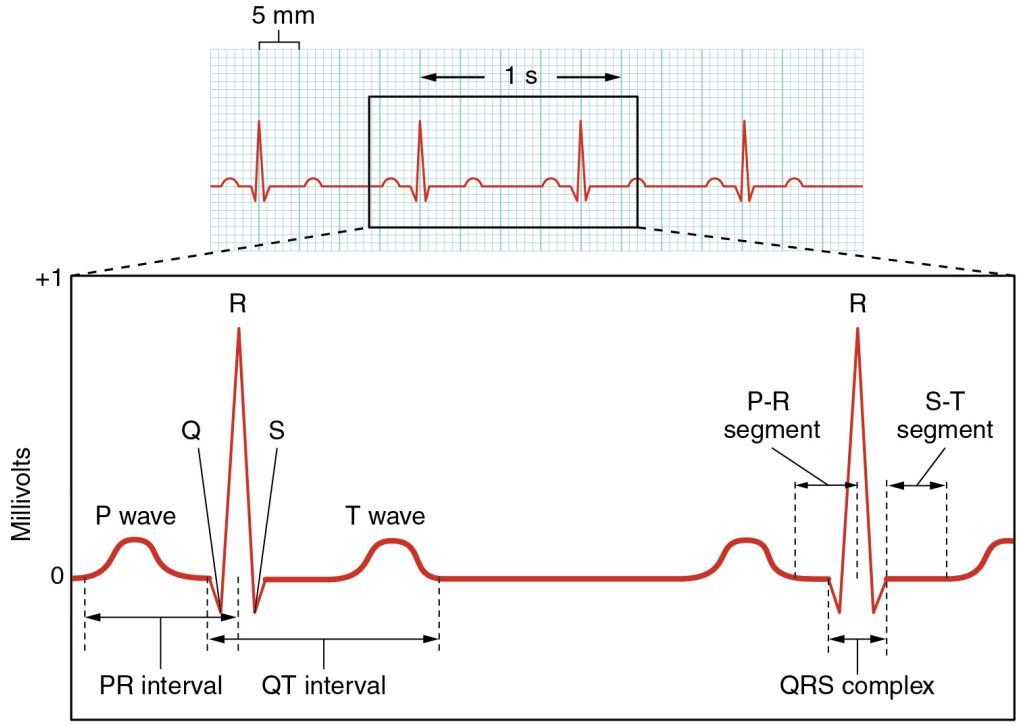
Normal ECGs include P waves, QRS complexes, and T waves [18, 114], where an ECG complex is composed of multiple ECG waves, *i.e.* the QRS complex is composed of the Q, R, and S wave. Usual morphological ECGs features comprise different waves and complexes, that come with different peak amplitudes and time lengths [48, 114]. Furthermore, ECG intervals and segments are among the commonly analyzed morphological features: a segment is the region between two waves, while an interval is a duration of time that includes one segment and one or more waves [18]. The most important waves, intervals, and segments are reported on a schematic ECG in Figure 2.2(a). The Table 2.1 summarizes the usual morphological features computed over ECGs with their physiological description and normal values for a healthy male adult, which were reported from Xie *et al.* [48].

In the following paragraphs we will describe the most important events related to the cardiac cycle, *i.e.* the beating activity of the heart from the beginning of one heartbeat to the beginning of the next, since each of the mentioned features is generated by specific cardiac events [18]. Thus, we anticipate a schematic representation of the heart in the Figure 2.2(b) for better understanding of the reader.

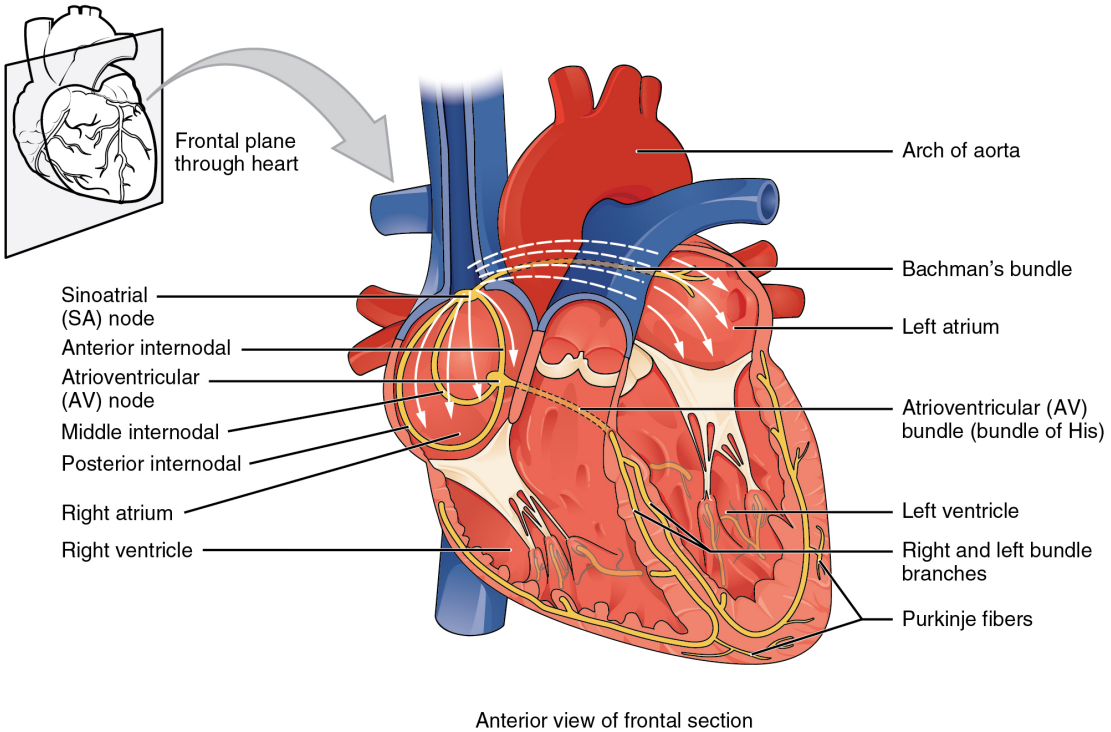
The depolarization of the Sinoatrial node (SA) occurs before depolarization of atrial myocytes[1], thus it anticipates the formation of the P wave. The SA node is located within the right atrium, thus its electrical impulses are difficult to be measured on the body skin surface. The stimulation of SA is conveyed to the right atrium and next to the left atrium, resulting in the generation of the P wave, that reflects the stimulation of the two atria. The P wave has a circular shape, an amplitude around $0.25mV$, and a length within $0.08 - 0.11s$. When the atrium enlarges, the conduction between the atria becomes aberrant, leading to P-mitrale or P-pulmonale waves [18].

The QRS complex depicts the propagation of an electrical stimulus across the ventricles. An entire QRS complex is composed of Q, R, and S waves. The R wave has wide amplitude, it has narrow length, and it represents the depolarization of the left ventricle. The average amplitude of a QRS complex is below $1.6mV$ at

[1]The muscle cells, including the cardiac ones, are also known as "myocytes" [4].



(a)



(b)

Figure 2.2: (a) Wave definitions and most important wave intervals and segments on a schematic ECG. Credits: OpenStax College, Creative Commons Attribution 3.0 Unported License, via Wikimedia Commons at `https://commons.wikimedia.org/wiki/File:2022_Electrocardiogram.jpg`. (b) A schematic illustration of the heart's anatomy. Credits: OpenStax College, Creative Commons Attribution 3.0 Unported License, via Wikimedia Commons at `https://commons.wikimedia.org/wiki/File:2018_Conduction_System_of_Heart.jpg`.

Table 2.1: The most important electrocardiogram waves, intervals, and segments along with normal values for a healthy male adult.

Features	Description	Amplitude	Duration
P wave	Atrial depolarization	0.25 mV	0.08-0.11s
P-R interval	Interval between the onset of atrial depolarization and the onset of ventricular depolarization		0.12-0.2 s
QRS complex	Ventricular depolarization	1.6 mV at the R peak	0.06-0.1s
ST-segment	The interval between ventricular depolarization and repolarization		0.05-0.15s
T wave	Ventricular repolarization	0.1-0.8 mV	0.05-0.25 s
U wave	Last phase of ventricular repolarization	<0.1 mV	Unknown
R-R interval	Interval among two consecutive R waves of the QRS complex ventricular rate		0.6-1.2s
Q-T interval	Interval needed for ventricular depolarization and repolarization		0.35-0.45s

the R peak, and its average duration is within $0.06 - 0.1s$. The count of QRS complexes occurred in a minute is typically used to calculate the heart rate. If the conduction between the left and right bundle branches of the heart is blocked, it may happen ventricular enlargement either hypertrophy, in which the QRS complex becomes wider, deformed, and prolonged in time [18].

The T wave can be observed after the QRS complex and it is generated by the repolarization process of ventricular myocytes. The T wave comes next to the QRS complex and it has an amplitude within $0.1 - 0.8mV$, and a duration of $0.05 - 0.25s$. A healthy T wave is positive and its assessment can be leveraged to diagnose several CA. For instance, it is frequent to see inverted T waves on patients with myocardial infarction or patient affected by pulmonary embolism [18].

The U wave is the final wave that can appear in the ECG which shows a circular upward deflection, with an amplitude lower than $0.1mV$. Typically, the polarity of a U wave is the same as the T wave, and because of their low amplitudes, U waves are not always visible and their generation processes are still not clear. Nowadays, U waves are assumed to represent the repolarization process of Purkinje fibers [129], which allow the heart's conduction system to create synchronized contractions of its ventricles, and they are essential for maintaining a consistent heart rhythm [4]. U wave inversions can be induced by myocardial ischemia or hypertension [129].

ECGs segments and intervals represent each stage of the cardiac cycle, that should be performed in a certain time frame under healthy conditions. An irregular cycle implies the presence of one or more CA [18].

The P-R interval represents the time between the onset of the P wave and the starting of the QRS complex. The P-R interval time length in normal ECGs is within $0.12 - 0.2s$, and it represents the diffusion of electrical conduction at the atrioventricular junction. A prolonged P-R indicates delayed conduction of the SA impulse to the ventricles, and it is usually referred as first-degree atrioventricular block. On the other hand, a short P-R interval can be seen when the atrioventricular node delay is bypassed, such as in the Wolff-Parkinson-White syndrome [18].

The ST segment represents the time elapsed from the end of the QRS and the beginning of the T wave. Since the myocytes of left and right ventricles are activated during ST segment, the contribution of the electric field vector generated by them is minimal in surface ECGs, and the ECG amplitude in the ST segment is slightly above the usual baseline level. Its typical duration is within $0.05 - 0.15s$, and the most common disorder related to the ST segment is Myocardial Infarction. This CA usually happens in the presence of prolonged ischemia which leads to the necrosis of part of the myocardium. Such event makes the difference of potential at ventricles still persist after depolarization, resulting in an ST segment drift on the ECG [18].

The R-R interval is the time passed between two successive R waves, and it is widely used to estimate the ventricular rate. Its usual values in healthy people are within $0.6 - 1.2s$. For instance, patients with atrial fibrillation show decreased average R-R, and increased R-R variability during sinus rhythm [18].

The Q-T interval is the time elapsed between the starting of the QRS and the conclusion of the T wave, reflecting the entire time necessary in ventricular depolarization and repolarization. The usual values of the Q-T interval are within $0.35 - 0.45s$. A prolongation of the QT interval has been observed in several CA associated with increased risk of sudden death [18].

2.1.3 Machine Learning within Electrocardiogram Classification

The first automatic interpretation computer programs were designed to classify CA from ECGs by setting thresholds and logic rules to replicate the physicians' reasoning process relying on the features we reported in Subsection 2.1.2 (and several others depending on the problem at hand) [23, 37, 47]. Next, researches experimented ML classification algorithms, in which the proper classification thresholds and rules were automatically learned from a set of training data [48, 114]. ML algorithms designed for classification return an output class relying on several handcrafted features which are provided as their input [44, 45]. We will refer to the above-mentioned ML methods as *standard* ML algorithms, since they follow the standard workflow

of ML, composed of feature calculation, followed by classification. Even if ML algorithms proved their effectiveness in classifying CA from ECGs, the process of feature engineering is sometimes difficult, time-consuming, and it heavily depends on human expertise [37, 38, 47].

In the last years researchers experimented *end-to-end* DL classifiers, even inspired by their success in other fields in classifying images, speech, texts, and several other kind of data [38, 51]. The end-to-end DL algorithms are a subset of ML algorithms that provide classification outputs relying on raw training data (or slightly preprocessed), without the need of designing handcrafted features, since they properly learn them directly from the input data [51]. Such algorithms gained more and more popularity in the context of ECG classification, and nowadays they represent one of the major research trends for addressing such task [38, 49, 54, 55].

In the next two subsections we will introduce some of the most employed standard ML and end-to-end DL algorithms in the context of ECG classification, along with the analysis of some of the wide number of research works that employed them.

Standard Machine Learning Algorithms

Among the most employed standard ML algorithms to classify CA from ECGs we report k-Nearest Neighbors (k-NN), Support Vector Machine (SVM), Random Forests (RF), and DNN [48, 114]. The mentioned ML algorithms rely on the standard workflow of ML, usually composed of noise reduction (Subsection 2.1.1), feature engineering (Subsection 2.1.2), and final training of the employed ML algorithm.

The k-NN algorithm is one of the most simple ML classification algorithms, that we mentioned in the Section 1.3 [44]. The k-NN algorithm represents input features as vectors in a multi-dimensional real valued space, along with a distance metric (*e.g.* the Euclidean distance, or any kind of L_p-norm). To classify a new input data, its computed features are reported onto the multi-dimensional space, and the output class is determined through majority voting on the classes of the k-closest feature points. Despite being easy to implement, the computational complexity

of k-NN rapidly grows when the dimensionality of features increases, due to the need of computing distances in high-dimensional feature spaces [44].

In the context of CA classification from ECGs, Park *et al.* [130] located heartbeats on ECGs through the detection of QRS complexes, by leveraging the Pan-Tompkins algorithm [131]. Then, the authors computed R-R time related features in the time-domain, and they classified CA from ECGs by employing k-NN as classifier. The algorithm was tested on the MIT-BIH dataset [127], and it achieved a classification sensitivity of 97% and specificity of 97%. Jung *et al.* [132] computed features using WT, and then they reduced their dimensionality through either PCA or LDA. Then, CA were classified from ECGs relying on k-NN, obtaining remarkable performance with sensitivity and specificity $\geq 95\%$ on the MIT-BIH dataset [127]. Venkataramanaiah *et al.* [133] classified ECGs into normal and abnormal beats. The authors detected heartbeats on ECGs by properly squaring and thresholding the signals, and they computed heart rate variability features. Then, the authors employed the k-NN algorithm to identify abnormalities in ECG beats, by obtaining a final classification accuracy of 99% on the MIT-BIH dataset [127].

SVM classifies feature points in a potentially high-dimensional linear feature space by building an hyperplane which separates them depending on their associated class [44, 45]. The hyperplane that allows for optimal separation within classes is the one that shows the maximum distance to the nearest training data point of each class (which is usually referred as functional margin). The larger is the margin, the lower will be the expected classification error of the trained SVM classifier on unseen data. Even if features may be represented in a linear space, it often happens that classes are not linearly separable in such space. Hence, the original features are often mapped onto a higher-dimensional space relying on kernel functions, with the assumption that their separation could be easier in such space [44]. It must be finally noted that SVM was originally designed for binary classification problems. However, in the case of multi-class classification the underlying problem is divided into multiple binary classification problems, where binary SVM classifiers are trained

per each pair of classes (thus obtaining $n(n-1)/2$ classifiers), or for a single class versus all the remaining ones (thus obtaining n classifiers) [44, 45].

Within the context of CA classification from ECGs, Yang *et al.* [134] computed features from ECGs by employing a custom DL architecture, namely the PCA Network, in which PCA is executed in cascaded stages to learn multi-stage filter banks. Finally, the authors classified CA from ECGs by setting features as input of a linear SVM classifier. The authors determined the effectiveness of the proposed technique on the MIT-BIH dataset [127], and they obtained an accuracy of 98%. Gliner *et al.* [135] classified ECGs into four categories: normal rhythm, atrial fibrillation, noisy segment, or other rhythm disturbances. The authors identified heartbeats by locating the R peaks through a custom detector presented by the same authors. Then, the authors computed time-frequency domain features, the average variability of the intra-beat temporal intervals, and the average morphology of heartbeats. The computed features were set as input of a SVM, and the authors obtained a F1-Score of 80% on the hidden subset of the 2017 PhysioNet Challenge dataset [136]. The introduced algorithm resulted in the top 25 positions in the final challenge leaderboard. Jha *et al.* [137] computed features from ECGs relying on the tunable Q-wavelet transform. Each ECG was decomposed up to the sixth level of the tunable Q-wavelet transform, and approximate coefficients at the sixth level were selected as features. Then, the authors classified CA relying on a kernel SVM with a radial Gaussian basis function as kernel. The average accuracy, sensitivity, and specificity offered by the proposed classifier on eight different classes contained in the MIT-BIH dataset [127] were respectively 99%, 96%, and 99%.

The RF are ML algorithms that build an ensemble of decision trees at training time, and then provide an output class that is the mode of the classes returned by the decision trees in the ensemble [44, 45]. Within RF, features with high discriminative capability are retained by the nodes of the trees in the process of the generation of the ensemble. Such ML algorithms come with several advantages, including low computational complexity, reduced overfitting with respect to a single decision tree, and unneeded data normalization prior to the ML training steps [44, 45].

In the context of classification of CA from ECGs, Vimal *et al.* [138] computed heart rate variability and DWT related features, and performed CA classification relying on a RF classifier, obtaining a classification accuracy $\geq 98\%$ on the MIT-BIT dataset [127]. Kung *et al.* [139] extracted a few important features from ECGs, including for instance the R-T interval and P-R interval, in order to design real-time classifiers to classify CA from ECGs. The authors employed RF classifiers to recognize Supraventricular Ectopic Beats (SEB) and Ventricular Ectopic Beats (VEB). The performance of the trained ML models reached F1-Score values of 81% for SVEB and 97% for VEB on the MIT-BIH dataset [127]. Rahul *et al.* [140] located heartbeats on ECGs by locating the QRS complex with a detection technique proposed by the same authors. Then, the authors extracted R-R interval features and statistical features from the heartbeats to classify ECGs within Normal, Premature Ventricular Contraction (PVC), and Premature Atrial Contraction (PAC) classes. CA were classified from ECGs by leveraging a wide set of standard ML algorithms, including RF which obtained an overall accuracy of 99% on the MIT-BIH dataset [127]. Yang *et al.* [141] computed a wide range of features on ECGs, including R-R intervals related features, WT features, HOS features (the authors divided the heartbeats into five intervals, and they computed skewness and kurtosis), and 1-dimensional local binary patterns (LBP)[2]. Then, the authors classified CA from ECGs by constructing an ensemble multi-class classifier composed of RF. The proposed method was trained and then evaluated on the MIT-BIH dataset [127], and the author obtained overall accuracy and average positive predictive value of respectively 98% and 94%.

DNN, or even referred as *Deep Neural Networks*, were already briefly introduced in Section 1.3, and they are ML classification algorithms vaguely inspired by the biological neural networks that compose the human brain [44, 51]. DNN are composed of an ensemble of connected nodes, usually called *neurons*, where

[2]LBP are features usually computed over 2-dimensional grayscale images [142], that the authors adapted to 1-dimensional ECGs. To compute LBP, an ECG window of size w is selected, then each sample point p in the window is compared with the central point of the window p_c. The value of p is then set to 1 in the LBP if $p > p_c$, otherwise, it is set to 0.

each connection, usually referred as *edge*, transmits a weighted signal to another neuron, similarly to what happens with the synapses into the human brain [44]. The magnitude of a specific weight increases or decreases the strength of the signal flowing through the respective edge. A neuron receives several signals and it processes them to submit to the other connected neurons. Such signals are represented by real numbers, and the output of each neuron is computed as the weighted sum of the input signals that come from other connected neurons. Finally, the computed sum is usually tresholded with a certain non-linear function to properly catch non-linearities from data (*e.g.* Sigmoid, hyperbolic tangent, or Rectified Linear Unit (ReLU) functions are usually employed) [44]. The weights are adjusted to fit to the problem at hand through the gradient descent algorithm, which requires the calculation of derivatives that are numerically computed through the backpropagation algorithm [143]. Neurons are typically aggregated into an input layer, an output layer which provides classification outputs, and a hidden layer between such two. When DNN are composed of multiple hidden layers, researchers usually refer to them in the literature with the terminology of DNN, instead of "artificial neural networks" [51]. Similarly to other fields, within the context of ECG classification DNN with more than one hidden layer are by far the most commonly employed [49, 55].

Sannino *et al.* [144] proposed a DNN composed of seven hidden layers and ReLU activation functions to classify CA from ECGs. The authors employed respectively 5, 10, 30, 50, 30, 10, and 5 neurons in each of the employed seven layers, and the employed loss function was the categorical cross-entropy. The ECGs were segmented into single heartbeats and set as input of the DNN, along with R-R interval related features computed in the time-domain. The authors obtained 99% of accuracy on the MIT–BIH dataset [127]. Bouaziz *et al.* [145] computed several features on ECGs, including ECG wave amplitudes, ECG intervals, WT related features, and standard statistical features. Then, the authors designed a DNN to classify CA from ECGs, and they applied a particle swarm optimization algorithm to optimize its weight, instead of the standard gradient descent algorithm. During the training step of the

DNN, particles were defined to be the matrices of weights that connected layers. The proposed model obtained an accuracy, sensitivity, and specificity $\geq 99\%$ when trained on the MIT-BIH dataset [127]. The experimental results suggested that the particle swarm optimization method was a valuable alternative to the usual gradient descent training algorithm for DNN that come with several drawbacks, including slow convergence and thus the possibility of being easily trapped in local minima [146]. Jothiramalingam *et al.* [147] detected Left Ventricular Hypertrophy (LVH)[3] from ECGs. The authors computed the location of R waves, S waves, inversion of the QRS complex, and variations in the ST segment by means of WT. A wide range of ML algorithm was employed to classify LVH from ECGs, including DNN. The accuracy in detecting LVH on the St. Petersburg INCART 12-Lead Arrhythmia Database [46] in the case the DNN was used was 98%.

Apart from the above-mentioned ML methods, there are even more ML classifiers that have been employed for ECG classification, such as fuzzy logic based classifiers [149], Gaussian mixture model based classifiers [150], ensemble models (including Bagging and AdaBoost) [151], and Bootstrap aggregating ensemble methods [152]. To deepen the wide range of employed ML algorithms in the context of classification of CA from ECGs the reader may check several recent survey papers, including for instance Berkaya *et al.* [114], Minchol *et al.* [47], and Xie *et al.* [48].

End-to-end Machine Learning Algorithms

A wide range of end-to-end algorithms based on DL has been used to classify CA from ECGs [49, 54, 55]. Within the most employed end-to-end DL algorithms we report Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long-Short Term Memory (LSTM), and auto-encoders. Unlike the standard ML algorithms, the mentioned end-to-end algorithms do not rely on the standard workflow of ML, but they classify ECG by directly leveraging raw ECG training data without the need of designing handcrafted features.

[3]LVH is a CA in which the left ventricle walls become thick due to prolonged hypertension, which may result in failing to pump blood effectively [148].

The CNN was briefly introduced in Section 1.3 where we presented three state-of-the-art works that classified CA from ECGs by leveraging it, *i.e* Hannun *et al.* [67], Ribeiro *et al.* [71], and Zhu *et al.* [70]. The CNN architecture is a kind of DNN that is capable of learning filters that applies convolutional operations to each sub-region of the input [44, 45]. In terms of structure, a CNN is typically composed of convolutional layers, pooling layers and fully connected layers. In convolutional layers, it is calculated a convolution of each sub-region of the input with a filter (which is a real-valued matrix of weights) to compute features from the input of the previous layer. Pooling layers are usually set behind convolutional layers and they allow for down-sampling of computed features, while selecting the most representative ones. After the convolutional and pooling layers, the computed features of each sub-region are flattened into a one-dimensional vector, which is set as input of a fully connected layer (which is usually a DNN), where input data are mapped into different classes. Then, the backpropagation algorithm along with the gradient descent algorithm [143] are leveraged to learn the optimal weights relying on a loss function, which in multi-class classification problems is usually selected as the categorical cross-entropy [44, 45].

Lai *et al.* [153] collected a dataset from 55 patients which were monitored for atrial fibrillation in around 24 hours by patch-based ECG machines positioned on upper-left chest, and standard 12-lead Holter devices. Two expert physicians annotated the dataset for atrial fibrillation and normal sinus rhythm, which the authors classified by means of four different CNN architectures. The authors set the raw ECG signal as input of one of the CNN models, which was composed of two convolutional layers, two pooling layers, and a final fully connected classification layer. Promising classification performance were obtained in terms of 93% accuracy, sensitivity, and specificity. Romdhane *et al.* [154] segmented heartbeats from ECGs contained in the MIT-BIH [127] and St. Petersburg INCART 12-Lead datasets [46], by defining an heartbeat to start at an R-peak and end after 1.2 times the median R-R time interval in a 10s window. Then, the authors trained a CNN to classify CA from ECGs by optimizing the focal loss to address the problem of classes imbalance.

The focal loss is a dynamically scaled cross-entropy where the scaling factor decays to zero as the confidence of the classification increases [105]. Intuitively, this scaling factor can automatically downweight the contribution of the frequent samples during training, thus allowing the DL model to focus more on hard samples. The CNN architecture consisted of two distinct convolutional blocks, composed of three convolutional operations which employed 256 filters. After each block, dropout [96] and batch normalization [155] were employed to regularize the training process, and a fully connected layer composed of 128 neurons was employed to achieve the final classification. The proposed CNN achieved overall performance of 98.41% accuracy, 98.38% F1-score, 98.37% precision, and 98.41% recall. Degirmenci *et al.* [156] segmented ECGs contained in the MIT-BIH dataset [127] into heartbeats relying on the WFDB Toolbox [157]. Then, the authors transformed the heartbeats into 2-dimensional grayscale images by directly plotting their time-amplitude waveform. Such images were set as input of a CNN architecture to classify CA, which was inspired by the LeNet architecture of LeCun *et al.* [158]. The CNN was composed of three convolutional layers respectively containing 64, 32, and 16 convolutional filters, followed by the same number of pooling layers. A final fully connected layer was leveraged to classify within 4 CA and normal sinus rhythm. The experimental results show that the classification performance of the proposed CNN reached 99.7% accuracy, 99.7% sensitivity, and 99.22% specificity.

The RNN is a kind of DNN that allows for recursion in the evolution direction of the input sequence, in which all the neurons are connected in a chain [159]. Each neuron in a RNN takes the input of the previous one and passes its output to the next neuron, thus making RNN able to obtain an output dependent only on its previous computation. This DNN architecture was shown to be effective in processing time series data [159]. However, the advantages of the traditional RNN often weakens when facing long-term dependencies [159]. Hence, to address this drawback LSTM networks were proposed by Hochreiter *et al.* [160], where each neuron of the traditional RNN networks is replaced with a memory unit. The core idea of LSTM networks is to update memory units continuously, allowing to store

relevant information while removing the redundant ones. According to the recent survey works of Ebrahimi *et al.* [55], and Liu *et al.* [49], the LSTM is the most popular kind of RNN architecture used within ECG classification so far.

Le Sun *et al.* [161] classified atrial fibrillation from ECGs by means of a stacked DL architecture composed of two hidden LSTM layers, with 55 memory units in each layer. The DL architecture received an input ECG composed of 100 time samples, and the authors trained it jointly onto the Long-term AF dataset [162] and the AF termination challenge database [163]. The dataset were composed of 84 long-term ECGs from subjects showing paroxysmal or sustained AF. Each ECG contained two simultaneously recorded signals sampled at $128Hz$. After training the LSTM architectures onto such datasets, the authors obtained 92% accuracy and 92% F1-Score in classifying atrial fibrillation versus normal sinus rhythm. Petmezas *et al.* [164] classified four ECG rhythms, namely normal sinus rythm, atrial fibrillation, atrial flutter, and atrioventricular junctional rhythm. The authors extracted features from raw ECGs by means of a CNN composed of three convolutional layers, each one followed by a pooling layer. Then, they performed the classification step by setting the extracted features as input of a LSTM model composed of 64 memory units, which was trained by optimizing the focal loss to deal with training data imbalance. The entire DL model was trained on the MIT-BIH Atrial Fibrillation dataset [165], and it achieved a sensitivity of 98%, and specificity of 99%, relying on a ten-fold cross-validation strategy. Chen *et al.* [166] classified CA from ECGs by combining two CNN architectures and a LSTM. The authors considered $10s$ ECG segments as input of the first CNN, and the computed R-R intervals on the same ECG segments as input of the second CNN. The outputs of the two CNN networks were merged to form a new input for a two-layer LSTM network, composed of 32 and 64 memory units respectively. The final DL model achieved 99% accuracy, under a five-fold cross-validation strategy on the MIT-BIH dataset [127]. Furthermore, the full DL architecture was validated relying on two additional independent datasets (The MIT-BIH Normal Sinus Rhythm Database [46], and the

MIT-BIH Atrial Fibrillation Database [165]), and it achieved an average accuracy of 97% when classifying normal sinus rhythm and atrial fibrillation.

Auto-encoders are DNN usually composed of an input layer, a hidden layer, and an output layer. The aim of auto-encoders is to encode the input to a lower dimensional representation in the hidden layer, and then decode it to recover the original input with the highest possible fidelity [167, 168]. An auto-encoder is thus trained to minimize the reconstruction error between the input and the recovered output. Usually, multiple auto-encoders are disposed in several layers in order to be capable of reconstructing complex input data, and they are usually referred in the literature as *stacked* auto-encoders [55, 168]. Usually, auto-encoders are employed as feature extractors for ECGs: the lower representations of the input signal that they store in hidden layers are used as input features of standard ML algorithms, such as a SVM or a DNN [49, 55].

Within the context of ECG classification, Hou *et al.* [169] designed an auto-encoder composed of two LSTM networks. The auto-encoder model was trained to reconstruct the considered ECGs in a first stage. After training, the weights associated with the neurons which composed the hidden layer of the auto-encoder architecture were used as features to classify CA from ECGs, by leveraging a SVM. The proposed method achieved average accuracy, sensitivity, and specificity of respectively 99.74%, 99.35%, and 99.84%, in a beat-based cross-validation approach, and respectively 85%, 63%, and 91%, in a record-based cross-validation approach, when trained on the MIT-BIH dataset [127]. Nurmaini *et al.* [170] leveraged auto-encoders by training them to reconstruct ECGs, similarly to the work of Hou *et al.* [169]. Then, the lower ECG representation of the hidden layer was set as input of a DNN which was employed as a classifier. The DNN classifier was composed of five layers, respectively composed of 32, 63, 126, and 5 nodes. The introduced classification model achieved an accuracy, sensitivity, specificity, precision, and F1-Score of respectively 99.34%, 93.83%, 99.57%, 90%, and 91% when trained on the MIT-BIH dataset [127]. Siouda *et al.* [171] classified CA from ECGs relying on an auto-encoder as feature extractor, and an ensemble of multiple neural networks

as a classifier. The original multi-class classification problem was decomposed into simpler binary classification sub-problems which were independently addressed by multiple DNN classifiers. To overcome the problem of imbalanced data, the authors applied the SMOTE algorithm [172] to add synthetic samples, according to the number of training instances in each sub-problem. The experiments performed on the MIT-BIH dataset [127] reported $> 99\%$ accuracy and showed that solving each sub-problem independently could enhance the accuracy, sensitivity, and specificity.

Apart from the above-mentioned DL based methods, there are even more DL classifiers that have been employed for ECG classification. To deepen the wide range of other employed DL algorithms in the context of classification of CA from ECGs the reader may check several recent survey research papers, including for instance Ebrahimi *et al.* [55], Liu *et al.* [49], and Somani *et al.* [54].

2.2 State of the Art on Machine Learning Explainability

In Section 2.1 we pointed out that the outcomes obtained by ML and DL models could potentially provide high performance in the task of supervised classification of CA from ECGs. However, as we already realized in Chapter 1, it is frequent to perceive such models as black-boxes, since insights about their functioning are mostly opaque for humans.

In this Section 2.2, we will first provide the essential definitions needed to introduce ML explainability for supervised classification ML models. Then, we will present a categorization useful to frame the available methodologies designed to explain supervised ML models, that we derived relying on both the survey works of Burkart *et al.* [76] and Guidotti *et al.* [75]. For each of the presented explainability categories, a few methodologies will be introduced with particular focus to some of the ones that will be employed in the next chapters. Finally, we will illustrate several research works that addressed the problem of ML explainability in the context of classification of CA from ECGs.

2.2.1 Preliminary Notions

In the present subsection, we provide a set of preliminary notions that will allow us to categorize the available techniques designed to explain supervised classification ML models.

A Supervised ML (SML) classification algorithm is trained to create a *model* $h(\boldsymbol{x}) = y$ which maps a real-valued input feature vector $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ to a target class $y \in \mathcal{Y} \subseteq \mathbb{R}$. To lean a proper classification model, the SLM classification algorithm must be fed with a training dataset $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$, where n is equivalent to the amount of available training data [44, 45]. SML algorithms are usually categorized depending on the underlying task they are employed to address, that usually are *classification* or *regression* [44, 45]. In the former, the target values $\mathcal{Y}$ are discrete, and they are usually referred as *classes* or *labels*. For instance, in the case of binary classification we have two target classes and usually $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{-1, 1\}$, while in the case of multi-class classification $\mathcal{Y} = \{0, \ldots, C\}$, where C is equivalent to the number of possible labels. The regression task aims to predict a real-valued target value $y \in \mathbb{R}$, which we will not consider since the vast majority of available ECG classification methods considers CA which are naturally mapped into discrete labels [47–49, 114].

A classification model provided after executing a SML algorithm can be a *black-box* $b : \mathcal{X} \to \mathcal{Y}$, $b(\boldsymbol{x}) = y$ where $b \in \mathcal{B}$, and $\mathcal{B} \subset \mathcal{H}$, which is the hypothesis space of all the black-boxes [76]. The hypothesis space $\mathcal{H}$ is the space of all the possible hypotheses for mapping inputs to outputs within SML algorithms are capable of searching. Usually, a specific SML algorithm is limited to search within a subset of $\mathcal{H}$, due to the characteristics of the problem at hand, and structure of the SML algorithm itself. For instance, in the case of black-boxes $\mathcal{B}$ could be equivalent to the set of artificial neural networks with two hidden layers. On the other hand, we can have an intrinsically explainable *white-box* model $w : \mathcal{X} \to \mathcal{Y}$, $w(\boldsymbol{x}) = y$ where $w \in \mathcal{I}$, and $\mathcal{I} \subset \mathcal{H}$, which is the hypothesis space of intrinsically explainable models for which we can immediately understand the

reasons behind their classification outputs [76]. For instance, in the latter case $\mathcal{I}$ could be equivalent to the decision trees of depth five.

To assess the classification performance of SML models after training them, it is leveraged an *error* measure, even referred as *loss*, or *score* $S : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ which is computed between the classification outputs and the ground truth labels. A simple example of error measure within binary classification is represented by the *hinge loss* $S(h(\boldsymbol{x}), y) = \max\{0, 1 - h(\boldsymbol{x}) \cdot y\}$. In the case $\mathcal{Y} = \{-1, 1\}$, the loss is equal to zero if the ground truth label y and the classification output $h(\boldsymbol{x})$ are equivalent [44, 45]. After selecting a specific error measure S, a SML algorithm can be defined as an optimization problem: given a training dataset $\mathcal{D}$, a SML algorithm attempts to solve the following equation

$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} S(h(\boldsymbol{x}_i), y_i), \tag{2.1}$$

in which the error measure S is averaged across all the training data and then minimized, and h^* is the optimal output SML model [44, 45]. Furthermore, several SML classification algorithms train parametric models $h(\boldsymbol{x}; \theta)$, in which θ is a vector of parameters. In such case, Equation (2.1) is adapted as follows [44, 45]:

$$h^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} S(h(\boldsymbol{x}_i; \theta), y_i). \tag{2.2}$$

In most of the cases, the optimization problems related to Equations 2.1 and 2.2 cannot be analytically solved due to their underlying structure. For instance, a common (and rare) example where an optimum solution can be computed is the case of linear regression [44, 45]. Thus, we usually obtain sub-optimal solutions found by means of numerical approximations, *e.g.* in the case of neural networks the parameters θ are computed relying on the gradient descent algorithm [44, 45].

Furthermore, the optimization problems defined in Equations 2.1 and 2.2 often lead to opaque black-box models for which it is required further inspection in order to explain the reasons behind their classification outputs. Indeed, researchers designed several ways to obtain explanations from black-box models which can be categorized as follows [75, 76]:

- Global *vs* Local explanations: at the highest level, approaches to gain explanations can be categorized into *model explanation* and *instance explanation* approaches [76]. The former methods provide insights about the internal functioning of the entire trained SML model, while the latter methods limit to explain the model output class y for a single input sample $\boldsymbol{x}$, or at most for its neighborhood. Model and instance explanation methods are often referred in the literature respectively as *global* or *local* explanation methods [75], and we will employ such latter terminology in the following sections for the sake of simplicity.

- Intrinsically explainable SML models: the most simple way to reach explainability is to employ SML models that can be explained on their own, since they have an intrinsically simple structure or they come with a limited number of parameters, thus being comprehensible to humans [75, 76]. By employing SML methods explainable *by design* there is no need of designing explainability approaches, since the employed SML model may be directly questioned to understand the reasons behind its classification outputs.

- Surrogate model explanations: an explanation for an outcome provided by a SML black-box may be provided by leveraging a surrogate intrinsically explainable SML model to globally or locally approximate the underlying black-box [75, 76]. The surrogate SML model may approximate the global functioning of the considered black-box model, or it may limit to proxy its behavior in the neighborhood of the sample of interest. Then, the explanation may be obtained by questioning the surrogate model, which is possible since it is intrinsically explainable by design.

Furthermore, explainability methods can be distinguished into *ante-hoc* and *post-hoc* explainability methods. In the former, the explanations are built-in in the SML model creation, while in the latter the explanations are provided only after the trained SML model is available [75, 76]. Explainability approaches can be either *model-agnostic* or *model-specific*, *i.e.* an explainability method can be

respectively suitable for all the kinds of SML models or it may be limited in explaining a specific one, or a specific class [75, 76].

In the following sections, global and local explanations, intrinsically explainable SML models, and surrogate model explanations will be further deepened. We will even introduce some well-known examples for each of the mentioned categories to highlight their principal aspects. To deepen more explainability approaches, we remark that literature review works which report far more further explainability methods are provided by Burkart *et al.* [76], Guidotti *et al.* [75], and Molnar [77].

2.2.2 Global and Local Explainability

A global or local explanation method e can be defined as a function

$$e : (\mathcal{X} \to \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y}) \to \mathcal{E}, \tag{2.3}$$

which considers a black-box SML model and a dataset as input, and it provides an explanation as an output, which belongs to the set of all the possible explanations $\mathcal{E}$ [76]. The explanation methods can be categorized into global and local methods [76].

Global explanation methods extract a global explanation for a black-box SML model b that is representative for a certain dataset $\mathcal{D}$, *i.e.* they compute $e(b, \mathcal{D})$. Usually, these approaches do not require the output class of the considered SML model to compute explanations and they only rely on the learned black-box model and the training data. In certain cases, the dataset is not even required to compute explanations [76].

The partial depencency plots (PDP) are an example of global explainability method which allow for *post-hoc* and model-agnostic explanations [77, 173, 174]. PDP show the average classification outcome of a SML black-box when a single feature (or a limited set) is varied over a certain range. The underlying idea of PDP is thus reporting how a certain set of features affect the final classification outcome of the considered SML model. To compute PDP, let $\boldsymbol{x} = \{x_1, x_2, \ldots, x_p\}$ represent the set of input features for a black-box SML model whose classification function

is $b(\boldsymbol{x})$. If we partition $\boldsymbol{x}$ into an interest set of features $\boldsymbol{z}_s$, and its compliment $\boldsymbol{z}_c = \boldsymbol{x} \setminus \boldsymbol{z}_s$, the *partial dependence* of the response on $\boldsymbol{z}_s$ can be calculated as

$$f_s(\boldsymbol{z}_s) = \mathbb{E}_{\boldsymbol{z}_c}[b(\boldsymbol{z}_s, \boldsymbol{z}_c)] = \int b(\boldsymbol{z}_s, \boldsymbol{z}_c) p_c(\boldsymbol{z}_c) d\boldsymbol{z}_c, \tag{2.4}$$

where $p_c(\boldsymbol{z}_c)$ is the marginal density of probability of $\boldsymbol{z}_c$: $p_c(\boldsymbol{z}_c) = \int p(\boldsymbol{x}) d\boldsymbol{z}_s$. Equation 2.4 can be estimared relying on a training dataset as

$$\bar{f}_s(\boldsymbol{z}_s) = \frac{1}{n} \sum_{i=1}^{n} e(\boldsymbol{z}_s, \boldsymbol{z}_{i,c}), \tag{2.5}$$

in which $\boldsymbol{z}_{i,c}$ $(i = 1, 2, \ldots, n)$ are the values of $\boldsymbol{z}_c$ available in the training dataset.

Computing a PDP relying on Equation 2.5 is rather immediate for the most of the black-box SML models [175]. For instance, let us limit $\boldsymbol{z}_s = x_1$ to be the unique feature of interest with associated the values $\{x_{11}, x_{12}, \ldots x_{1k}\}$. The PDP of the response on the feature x_1 can be computed as follows:

- For $i \in \{1, 2, \ldots, k\}$:
  - Make a copy of the training dataset and replace the original values of x_1 with the constant value x_{1i}.
  - Compute the outcomes relying on the modified copy of the training dataset.
  - Compute the average estimate $\bar{f}_1(x_{1i})$.
- Plot the pairs $\{x_{1i}, \bar{f}_1(x_{1i})\}$ for $i \in \{1, 2, \ldots, k\}$.

The calculation of PDP is rather intuitive, since the partial dependence for a particular feature value represents the average prediction in the case we force all the training samples to assume that value. Furthermore, computing PDP is often not computationally intensive since we do not require to train the underlying black-box, and we may compute PDP for a wide range of black-boxes since the method is not dependend on the structure of the selected SML model. However PDP come with two main drawbacks, where the first one is represented by the fact that the realistic maximum number of features which is possible to plot is three, being limited by

our inability to proper visualize more than 3 dimensions [77, 173, 174]. Finally, within PDP it is assumed that the features for which the partial dependence is calculated are not correlated with other features. In the case it happens, during the estimation process we may consider a wide set of points within the feature distribution for which their actual probability is very low [77, 173, 174].

Local explanation approaches provide explanations limited to a single input sample $\boldsymbol{x}$ and its respective output class y, and they cannot be generalized for understanding the entire working of a SML black-box model [76]. A simple example of model-specific, *post-hoc*, and local explanation method is represented by the process of looking at the decision paths when classifying an input sample $\boldsymbol{x}$ with a decision tree [76]. Other more complex approaches are represented for instance by counterfactual explanations [75, 76].

Counterfactuals are described by the Cambridge Dictionary of Psychology as: "thinking about what did not happen but could have happened" [176] and they can be formally expressed as follows: if x had been x', y would have been y'. Thus, a counterfactual refers to a different reality in which some other facts would have lead to different outcomes. The factual x comes with the associated consequence y, but if x changes to its counterfactual x', the consequence thus changes to y'. Counterfactuals have been employed as a kind of model-agnostic or model-specific, *post-hoc*, and local explanation method for SML black-boxes, depending on their specific implementation [75–77]. In the ML domain, x and x' are considered as inputs for a black-box, and y or y' are the respective classification outputs provided by it. In this case, the problem of exhibiting a counterfactual explanation becomes a search problem in the feature space where the aim is to find a counterfactual sample that leads to output a different class. Once the counterfactual is found, it may be presented either by itself or by highlighting the differences from its factual, to understand the variations responsible for the different classification outcome.

Several implementations of counterfactual explanations were proposed in the context of ML [75–77]. The most simple way for obtaining counterfactuals is by trial and error, *i.e.* by randomly changing the feature values of a considered

sample and stopping when the desired output gets classified [77]. Wachter *et al.* [177] computed counterfactual explanations by searching counterfactual samples as close as possible to the original ones so that a new output class is selected. The authors measured the distance in terms of the Manhattan distance [178], weighted by the inverse median absolute deviation. The Optimal Action Extraction (OAE) method can be leveraged in the case of RF classifiers, AdaBoost, and gradient boosted trees. The OAE approach tries to build a feature vector so that the desired output is obtained at minimum cost [179]. Finally, The Feasible and Actionable Counterfactual Explanations (FACE) method aims to find counterfactuals relying on the concept of shortest path distance defined trough density-weighted metrics [180].

The counterfactuals come with several advantages [75]. The understanding of a counterfactual explanation is relatively straightforward: if a certain feature value of a test sample is changed as it happens in the counterfactual, the classification outcome changes as expected. The counterfactuals are *post-hoc* methods and they do not usually require any access to the internals of the considered SML black-box. Futhermore, they can be adapted to a wide range of SML models due to their frequent model-agnostic property. However, a main drawback of counterfactuals is represented by the fact that for each instance we may find several counterfactual explanations, sometimes even contradicting within each other, and thus increasing the complexity of understanding them [77].

2.2.3 Intrinsically Explainable Models

A few SML models are not black-boxes, thus they belong to the hypothesis space $\mathcal{I}$, and they are explainable on their own due to their simple design or their limited number of trainable parameters [75, 76]. As a consequence, we can gain explanations by modifying Equation 2.1 and employing a white-box model as

$$w^* = \underset{w \in \mathcal{I}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} S(w(\boldsymbol{x}_i), y_i), \tag{2.6}$$

with $(\boldsymbol{x}_i, y_i) \in \mathcal{D}$. By solving Equation 2.6, we may learn a white-box model from the hypothesis space of the white-box models $\mathcal{I}$, where justifications about classification outputs can be easily derived by querying the model itself.

The intrinsically explainable models provide *ante-hoc* explainability, since the explanations are built-in in the model training process. Furthermore, they are model-specific by design since the explanations are derived relying on the specific functioning of the underlying SML model. Among the most known approaches for obtaining intrinsically explainability we may find linear models, and decision trees [75, 76].

SML linear models are parametric models composed of input features and weights (the parameters) associated to each of them [44]. A certain weight immediately highlight the contribution of the respective feature to the final classification outcome. SML linear models are also often leveraged to yield continuous values instead of categorical class labels, thus being often employed in regression tasks. Potentially, it holds that for any parametric SML model we can analyze its parameters to determine their contribution in the final output, however it is unfeasible for several kinds of SML algorithms which produce highly parametric models, *e.g.* huge DNN. Usually, the more the model is parametric, the more the perspective of gaining this kind of explainability tends to vanish [75, 76].

Decision trees are SML models composed of tree nodes and leaf nodes [44]. The former are responsible to split features relying on a certain threshold value, while the latter output a class label. The classification process for an instance $\boldsymbol{x}$ starts at the top of the decision tree, and it proceeds downwards until a leaf node is reached. At every intermediate node, a certain feature is compared to the splitting threshold, and relying on the outcome of this comparison, the traversing process proceeds toward the left or the right part of the three. Decision trees are usually build following a greedy top-down approach, such that once a feature and a threshold are selected as a splitting criterion, they cannot be switched by another splitting feature and associated threshold [44].

The main advantage of relying on white-box models is to have explainability by design, that is desirable for fields in which understanding the reasons behind ML decisions is of paramount importance [75]. However, even if intrinsically explainable models could be highly required in several fields, they usually come

at the cost of performance, which is usually superior for black-box SML models [75, 77]. Thus, the main drawback of white-box models is represented by their simplicity in terms of limited number of parameters, which does not often allow to catch complex relationships from training data [76]. Intrinsically explainable models are then often not leveraged in the case high classification performance is required, or when the demand of explainability is not deemed as strictly necessary in the considered domain [75].

2.2.4 Surrogate Model Explanations

A common way of getting explanations from a black-box SML model is to use a white-box surrogate model to proxy the behavior of the black-box, thus allowing to understand its global or local functioning [75–77]. Formally, the process of surrogate model fitting is executed by approximating a black-box SML model with a white-box by solving the following equation

$$w^* = \underset{w \in \mathcal{I}}{\operatorname{argmin}} \frac{1}{\mid \mathcal{X} \mid} \sum_{x \in \mathcal{X}} S(w(\boldsymbol{x}), b(\boldsymbol{x})), \tag{2.7}$$

in which the error measure S is a *fidelity* score, *i.e.* it measures how well the surrogate white-box SML model w is capable of approximating the underlying black-box SML model b.

Surrogate models are designed under a *post-hoc* fashion and they are agnostic with respect to the kind of SML model to be explained [75–77]. Furthermore, they can be divided into global and local surrogate models depending on their level of approximation of the underlying black-box SML model [75–77].

A global surrogate model w proxies the black-box b on the entire training dataset $\mathcal{D}$, *i.e.* $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$. If the dataset far exceeds the computational resources at hand, the set $\mathcal{X}$ is sampled from the original training dataset to represent it sufficiently well. Thus, global surrogate models *mimic* the classification outputs of a certain black-box with comparable accuracy (when possible). Decision trees are usually employed to proxy the behavior of black-box models [75–77]. For instance, several research works designed global surrogate explanation models by training

a decision tree on a specific dataset $D' = \{(\boldsymbol{x}_1, b(\boldsymbol{x}_1)), \ldots, (\boldsymbol{x}_n, b(\boldsymbol{x}_n))\}$, *i.e.* the surrogate model is trained to mimic the predictions $b(\boldsymbol{x}_i)$ of the black-box SML model [76]. Hinton *et al.* [181] followed this approached and they introduced a decision tree that was trained with stochastic gradient descent relying on the predictions of a DNN. The decision tree surrogate model employed the learned filters to provide hierarchical decisions relying on input samples. Yang *et al.* [182] introduced a binary decision tree that was capable of catching the most relevant decision rules that were implicitly contained in a DNN black-box. The tree was trained on an input matrix built relying on the contributions of input features to predicted scores for each single prediction. For training the surrogate tree, the input feature space was recursively partitioned by maximizing the difference in the average contribution of the split feature between the divided spaces.

The fact that surrogate models are designed under a *post-hoc* fashion represent a great advantage since it is not required to train the underlying black-box to obtain explanations. Furthermore, the model-agnostic property allows to apply a wide range of surrogates to a black-box SML model depending on the problem at hand, without any major concern related to its structure [75, 76]. However, when using surrogate models researchers must be aware that the obtained explanations are often focused only on the considered black-box, and they do not account for the ground truth associated to the training data, since the surrogate model never sees the real outcome (as it happens in Hinton *et al.* [181]). Furthermore, even we can measure how close the surrogate model is to the black-box SML model, it could happen that the white-box model is very close for one subset of the dataset, but widely divergent for another one [77].

A local surrogate model w proxies the black-box b in the neighborhood of a sample $\boldsymbol{x}$, defined as $\mathcal{X} = \{\boldsymbol{x}' \mid \boldsymbol{x}' \in N(\boldsymbol{x})\}$, where N is a certain function that defines the neighborhood of $\boldsymbol{x}$ [75–77]. Local surrogate models are only accurate in the neighborhood of the current prediction, and they allow only for a local understanding of black-box SML models. A widely known example of local surrogate method is

represented by the Local Interpretable Model-agnostic Explanations (LIME) [183], which will be presented in the details in the next subsection.

2.2.5 Local Interpretable Model-agnostic Explanations (LIME)

In the present subsection we introduce the LIME explanation algorithm [183], which is a model-agnostic, *post-hoc*, and local surrogate explanation algorithm which falls in the class of Local Linear Explanations (LLE) [184], that is also referred in the literature as *feature importance* models [75].

To define LLE let us consider a sample $\boldsymbol{x} \subseteq \mathbb{R}^F$ that is set as input of a black-box b. The black-box model b does not come with any kind of assumption on the LLE structure. A LLE function g explains the prediction of the black-box $b(\boldsymbol{x})$ by training a white-box classifier that mimics the black-box b in the neighborhood of the input sample $\boldsymbol{x}$, and it is defined as a linear function which takes the input sample $\boldsymbol{x}$ as

$$g(\boldsymbol{x}) = w_0 + \sum_{i=1}^{F} w_i \cdot \boldsymbol{x}^{(i)}. \tag{2.8}$$

The LLE g multiplies each feature $\boldsymbol{x}^{(i)}$ for a weight w_i to proxy the behavior of the black-box b in the local neighborhood of $\boldsymbol{x}$. The absolute value of the weight w_i provides the importance each feature $\boldsymbol{x}^{(i)}$.

The LIME algorithm build an LLE model g relying on an artificial neighborhood $N(\boldsymbol{x})$ built in the close vicinity of the input sample $\boldsymbol{x}$ for which it is required an explanation. A neighborhood $N(\boldsymbol{x})$ composed of H points around the input sample $\boldsymbol{x}$ is defined as

$$N(\boldsymbol{x}) = \{\boldsymbol{x}_j = \boldsymbol{x} + \boldsymbol{\epsilon}_j, \boldsymbol{\epsilon}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{\mathcal{X}}) \mid j = 1, \dots H\} \tag{2.9}$$

where the vectors $\boldsymbol{\epsilon}_j$ locally perturb the input sample $\boldsymbol{x}$ and they are sampled by a Gaussian distribution with zero mean and $\boldsymbol{\sigma}_X$ standard deviation, composed by the standard deviations of features in the training dataset[4]. To find the LLE g,

[4]In the case features are categorical, they can be uniformly sampled relying on their frequency in the dataset $\mathcal{X}$.

the LIME algorithm trains a Ridge regression model on the neighborhood $N(\boldsymbol{x})$ relying on the following linear least squares loss function

$$\mathcal{L}(b, g, \pi_{\boldsymbol{x}}) = \sum_{\boldsymbol{z} \in N(\boldsymbol{x})} \pi_{\boldsymbol{x}}(\boldsymbol{z})(b(\boldsymbol{z}) - g(\eta(\boldsymbol{z})))^2, \tag{2.10}$$

where the predefined kernel distance is set to $\pi_{\boldsymbol{x}}(\boldsymbol{z}) = \exp\left(\frac{-d(\boldsymbol{x},\boldsymbol{z})^2}{\gamma^2}\right)$ with kernel width $\gamma = \frac{3}{4}F$, and $d(\cdot,\cdot)$ is the standard Euclidean distance. The explanations can be provided in a space $\boldsymbol{X}'$ that could be different with respect to $\boldsymbol{X}$ if it is provided a proper mapping function $\eta : \boldsymbol{X} \rightarrow \boldsymbol{X}'$. The function η is set to the identity function in the case $\boldsymbol{X} = \boldsymbol{X}'$. The model complexity $\Omega(g)$ of the explanation model g can be added to Equation 2.10, which represents the number of features used by the explanation model g. The latter term can be not considered in the case the model g exploits the full set of features.

Intuitively, the LLE model g is a local explanation for the instance $\boldsymbol{x}$ because it classifies the set $N(\boldsymbol{x})$, which is an artificial dataset sampled around $\boldsymbol{x}$. The provided explanation is linear since it provides a single real-valued weight for each feature of the input sample $\boldsymbol{x}$. Finally, we notice that different kernels can be selected, but they potentially alter the provided explanations.

2.2.6 Explainability within Electrocardiogram Classification

In the previous sections we provided the required definitions for introducing the concept of ML explainability for SML black-box models, along with a categorization useful to frame the available explainability methodologies. Being aware of the concepts we introduced in the previous sections, we now illustrate the recent research works that addressed the problem of ML explainability in the context of classification of CA from ECGs.

Goodfellow *et al.* [185] developed a CNN to classify single-lead ECGs within normal sinus rhythm, atrial fibrillation, or other kind of CA. The proposed CNN was composed of 13 layers, including convolutional and pooling layers, ReLU activation functions [45], batch normalization [155], dropout [96], Global Average Pooling (GAP) [186], and a last fully connected classification layer. The GAP layer turns

a set of feature maps into scalar values by computing the average of the values contained in the maps, and after GAP is performed scalars are fed into the final fully connected layer. The authors trained the model on the 2017 Physionet Challenge dataset, which was composed of of 12,186 labeled ECGs [136], and they obtained average scores of 84% precision, 85% recall, 84% F1-Score, and 88% accuracy. Then, Class Activation Mapping (CAM) [186] was leveraged to understand which areas of the input ECGs the model was focusing on when performing a classification. CAM is a model-specific, *post-hoc*, and local explainabily method which takes the weights computed after the GAP layer, and the class for which we want to obtain explanations. Then, CAM considers the feature maps that pass through the GAP layer, it multiplies them with their respective weights, and it finally adds them together. The final weighted sum returns a heatmap for a specific class, which has the same size of the feature map, and provides hints about where the CNN paid attention when performing classifications. The authors finally observed that for normal sinus rhythm the general CAM pattern showed roughly constant attention, while in the case of atrial fibrillation the general CAM pattern showed random attention to several fluctuations within the ECGs. Finally the CAM related to other CA showed attention spikes mostly associated with the presence of premature beats.

Hicks *et al.* [187] introduced an approach called ECG Gradient Class Activation Map (ECGradCAM), which the authors leveraged to build explainable attention maps. The presented approach is similar to the one of Goodfellow *et al.* [185], since the authors relied on a modified version of the CAM explainability method, which is Grad-CAM [188]. In CAM, feature maps are weighted by multiplying them by the weights taken out of the GAP layer. On the other hand, in Grad-CAM, feature maps are weighted by computing "alpha values" which are calculated only relying on gradients, without the need of the GAP layer. Hence, the advantage of Grad-CAM is represented by the fact that the method does not require a specific DL architecture, because gradients can be obtained regardless the presence of the GAP layer. The authors slightly modified the GradCAM approach to compute heatmaps for each lead of the ECG, and they designed a CNN with residual connections

inspired by He *et al.* [73], with eight residual connections. The authors employed the designed CNN architecture for two different ML tasks: 1) to predict common intervals and waves amplitudes from ECGs in a supervised regression strategy, *e.g.* the QT interval, QRS duration and T wave amplitude. 2) To classify the sex of the patient from which the ECG was sampled. The authors trained the CNN in both the tasks under a five-fold cross-validation relying on the GESUS dataset [189], in which 7,152 samples were employed for training set and 1,787 for validation set. In the first case, the results provided by the authors in terms of mean absolute error suggested that the model was capable of predicting interval widths and wave amplitudes. Furthermore, explainability results suggested that the CNN properly inspected the right waves and intervals that were related to the predicted variable. In the second case, the authors obtained an accuracy of $\sim 88\%$, and the explainability results suggested that the QRS complex and the downslope of the R wave were the most relevant part of ECGs when distinguishing between a male and female patients. However, we must notice that 7 and 8 test samples were respectively analyzed to draw the presented conclusions in the regression and classification tasks, and a systematic assessment of the CNN model on the entire training dataset was not provided. As a remark, we notice that the work from Hicks *et al.* [187] it is not strictly related to the context of CA classification from ECGs. However, we reported the work for completeness of the literature review proposed in this section, since part of the research work was focused on a classification task whose technical background was analogous to the one employed within CA classification. Finally, we mention that a few other works employed GradCAM to explain the classification outcomes of a CNN when classifying CA from ECGs, and an example is reported by Vijayarangan *et al.* [190], which work is analogous to the one of Goodfellow *et al.* [185].

Several researchers explained the outcomes of black-boxes which classify CA from ECGs by means of attention based models [76], which are model-specific, *ante-hoc*, and local explainability methods. The attention mechanisms can be leveraged to highlight the most relevant parts of the input signal that lead to a

specific classification output. Yao *et al.* [191] introduced a CNN along with an explanation method based on an attention mechanism with the aim of adding explainability to the black-box CNN model. The attention mechanism provided the signal segment of interest along with classification result, and it was implemented as a DNN. The authors trained the CNN model on the China Physiological Signal Challenge dataset [192], which was composed of 6,877 12-lead ECGs from $6s$ to $60s$, labeled with a total of 8 kinds of CA and normal sinus rhythm. The final reported overall classification results obtained by the CNN model were 83% precision, 80% recall, and 81% F1-Score. Finally, only three ECG samples with PAC, PVC, and atrial fibrillation were drawn to analyze explainability results. Even if a systematic assessment of the CNN model on the entire training dataset was not provided, the few analyzed ECGs suggested that for PAC and PVC, larger weights were clearly assigned for segments showing premature beats. In the ECG showing AF, weights were more uniformly assigned since the distortions in ECG rhythm consistently appeared during the entire recording.

Mousavi *et al.* [93] proposed a hierarchical model to classify atrial fibrillation from normal sinus rhythm. The model was composed of three sub-networks in which each network was composed of a stacked bidirectional recurrent neural network [193]. Each sub-network was followed by an attention model capable of providing multi-level explainability, by considering segments within heartbeats, the whole heartbeat and the combination of all the heartbeats. The method was trained on a combination of the MIT-BIH Atrial Fibrillation Database [165] and the PhysioNet Challenge 2017 dataset [136], and it achieved average performance of 99.08% sensitivity, 98.78% specificity, and 98.83% accuracy. Regarding explainability, a few ECGs containing AF and non-AF categories were analyzed, suggesting that the proposed method paid attention to the irregularity of R-R intervals and the absence of P-waves, which are relevant clinical traits when diagnosing atrial fibrillation. Similar explainability works relying on attention mechanisms were proposed by Baalman *et al.* [92] and Hong *et al.* [194] *et al.*. The first authors built a DNN to classify ECGs within atrial fibrillation and normal sinus rhythm, along with an

DL based attention mechanism to allow for explainability. Through the attention mechanism the authors computed a heat map on the input signal to show the areas of the ECG that were mostly employed by the DL classifier to come to the correct classification. Finally, Hong *et al.* [194] developed a multi-level attention model by deepening the previous approach in extracting multilevel domain knowledge features, in terms of beat, rhythm and frequency domain level features.

Strodthoff *et al.* [91] investigated both CNN as well as RNN architectures to classify within normal sinus rhythm, anterior, and posterior myocardial infarction. The proposed CNN architecture was inspired by Long *et al.* [195] and it was composed by six convolutional layers, while the RNN was inspired by the ResNet architecture [73], and it was designed with three residual blocks. The authors trained both the DL models on the PTB database which is composed of 549 labeled ECGs from 290 subjects [46, 65]. In the case of the CNN, the authors obtained overall 93% sensitivity, 90% specificity, and 94% positive predictive value, while in the case of the RNN they obtained 92.5% sensitivity, 90% specificity, and 94% positive predictive value. The authors applied a model-specific, *post-hoc*, local explainability method named "gradient $\times$ input" [196] to identify which part of the input ECGs was the most relevant to the final classification. The gradient $\times$ input method computes a heatmap by calculating the signed partial derivatives of the output with respect to the input, and it multiplies them with the input itself. Different from what usually happens in computer vision, where the attributions of all color channels are added up together, the authors retained the different attributions of each channel to be able to highlight channel-specific effects. Regarding explainability, the authors assessed a few test ECGs and they observed that the ECG areas which most contributed to the final classification outcomes did not always align with the ones that physicians would have identified as important.

Zhang *et al.* [94] classified CA within 9 different classes (including atrial fibrillation and myocardial infarction) in addition to normal sinus rhythm. The authors designed a CNN with residual connections inspired by He *et al.* [197], and they employed four residual blocks. The introduced CNN accepted as input raw

12-lead ECGs, with a duration of $30s$ and sampling rate of $500Hz$, and it was trained on the China Physiological Signal Challenge dataset [192]. The final model achieved an average F1-Score of 81%, and its classification outputs were explained relying on the SHapley Additive exPlanations (SHAP) method [198] to interpret the CNN behavior at both the single ECG level and at the whole dataset level. The SHAP method is a LLE method which is model-agnostic, and *post-hoc* and it builds local and linear explanations relying on the concept of Shapley values [199]. Such values are computed relying on the coalitional game theory, by assuming that each feature value of the considered input instance is a player in a game where the classification output is the payoff [200]. The more a player is important in obtaining the output, the more it finally deserves the payoff, and Shapley values illustrate how to dispense such payoff among the players. The proposed work is interesting since it is the only one, to the best of our knowledge, which computed explanations by taking into account the entire dataset, and thus not limiting to a single ECG at a time. However, the dataset level explanations are sometimes difficult to analyze by the point of view of physicians since several of the considered CA may be observed on each of the 12 leads. For instance, classifying atrial fibrillation and atrioventricular block requires to even visualize P waves and P-R intervals and these findings can be assessed on all the 12 leads [50, 95], but for some reason the network focused its attention only on II, V1, and V2 leads. The same holds for myocardial infarction which can be observed on a wide range of leads, depending on its anatomical localization [112]. Finally, Shapley values were even employed by Ibrahim *et al.* [201] which highlighted the features that mostly contributed to the classification of myocardial infarction for an Extreme Gradient Boosting (XGBoost) model [45], demonstrating the high impact of age, sex, and QRS duration. The authors only explained two sample ECGs, however the reported results suggested that the network was often relying on not clinically relevant features when classifying myocardial infarction.

3

Design of Machine Learning Algorithms for Classification of Cardiac Abnormalities

Contents

3.1 Introduction

In the present Chapter 3 we will introduce our research works which were focused on the design of ML algorithms in the classification of CA from ECG signals.

In Section 3.2 we will present our research work where we designed an ensemble ML classification algorithm to classify 27 CA from ECGs [109]. Each classification model in the ensemble was trained on the 2020 PhysioNet/Computing in Cardiology challenge dataset [68]. Differently from most of the previous studies which employed DL, and often simply borrowed DL architectures from other domains, we designed an ensemble approach where each model in the ensemble was designed to specifically classify a subset of CA that altered the same set of ECG physiological features.

In Section 3.3 we will present our research work where we experimented three different Automated ML (AutoML) frameworks to address the time-consuming problem of automatically finding optimal ML and DL pipelines, to classify within 30 CA from ECGs [110]. The AutoML frameworks were trained on the 2021 PhysioNet/Computing in Cardiology challenge dataset [69], which is an extension of the previous year challenge dataset [68]. Cost-sensitive learning was leveraged to address imbalanced classes within the available dataset: we run the AutoML frameworks to train the underlying ML and DL models by minimizing a custom misclassification score, instead of the commonly employed ML and DL loss functions.

3.2 Classification of 12-lead Electrocardiograms with an Ensemble Machine Learning Approach

3.2.1 Introduction

In the context of Bodini *et al.* [109] we took part to the 2020 PhysioNet/Computing in Cardiology challenge, which asked participants to perform the automatic classification of 27 CA from 12-lead ECGs. We investigated on a hybrid classification approach, combining average-template-based algorithms with Convolutional Neural Network (CNN) models, to build an ensemble classification model. We calibrated

the model on the available 43,000+ ECGs, while organizers tested the model on private validation and test sets.

Standard ECG preprocessing was fist applied. For ECGs related to CA altering the ECG morphology, multi-lead average P wave, QRS complex, and T wave were computed. For signals associated with irregular rhythms, time dependent features were computed on the inter-beat time interval series (R-R). The ensemble model comprised of: 1) three CNN models to classify morphology-related CA. 2) a fully connected neural network to classify rhythm-related CA. 3) A threshold-based classifier for premature ventricular beat detection. The final classification output was obtained by combining the outputs of the classifiers.

On our validation set (derived from public training data), the ensemble model obtained class-wise recall values ranging between 71% and 93%. The highest and lowest recall values were respectively achieved by CA affecting the QRS complex, and rhythm. The organizers designed a score for ranking the models, and the ensemble model proposed by our team "BiSP Lab" reached the 40^{th} position when tested on the private test set, suggesting that our model showed potential for classification of CAs from ECGs.

3.2.2 The 2020 PhysioNet/Computing in Cardiology Challenge Dataset

We employed the publicly available subset of the dataset prepared for the 2020 PhysioNet/Computing in Cardiology challenge [68]. The entire challenge dataset was composed of a public subset shared for training algorithms, and a private subset which organizers employed to assess the instances which competed to win the challenge. The organizers divided the private data into validation and test sets. The entire dataset was composed of a total of 66,361 12-lead ECGs multi-labeled with one or more CA among 111 possible ones, while the classification results were scored by the challenge committee relying only on a subset of 27 CA. The publicly available subset of the dataset was provided with 43,101 recordings and the same set of 111 possible labels. We relied on such public subset and we did not consider

the labels not scored by the challenge organizers. When training the designed classification algorithms, we merged the classes complete right bundle branch block, premature atrial contraction, and premature ventricular contractions respectively with right bundle branch block, supraventricular premature beats, and ventricular premature beats since they were scored as the same diagnosis by the challenge committee. Thus, our ensemble model provides 24 output CA classes.

The publicly available dataset was sourced by merging six datasets obtained from several institutions located in four countries, across three continents:

- **CPSC2018** and **CPSC-Extra**. The first source is the China Physiological Signal Challenge 2018, held during the 7$^{\text{th}}$ International Conference on Biomedical Engineering and Biotechnology in Nanjing, China [192]. This source provided the public training dataset employed during the challenge (CPSC2018), and a further dataset that was disclosed after its conclusion (CPSC-Extra). Both the datasets were acquired from 9,458 patients.

- **INCART**. The second source is the St. Petersburg INstitute of CARdiological Technics (INCART), St. Petersburg, Russian Federation [46], which provided the INCART 12-lead Arrhythmia Database. The database was acquired from 32 patients, and it is available online on the website of the PhysioNet project.

- **PTB** and **PTB-XL**. The third source is the Physikalisch-Technische Bundesanstalt (PTB), the national metrology institute of Germany which is located in Brunswick, Germany. This source provided two publicly available datasets acquired on 19,175 patients: the PTB Diagnostic ECG Database [65, 66] and the PTB-XL Database [64]. A wider introduction of the PTB-XL dataset was previously provided in the Section 1.3.

- **G12EC**. The fourth source is the Emory University, Atlanta, Georgia, United States, which provided the Georgia 12-lead ECG Challenge (G12EC) dataset [68]. The latter is a novel dataset that was disclosed during the 2020 PhysioNet/Computing in Cardiology challenge, and it represents a large population from the southeastern United States, composed of 15,742 patients.

A summary of the ECG datasets provided by the sources we reported in the above list, which are contained in the public subset of the challenge dataset, is available in Table 3.1. In the table we reported the number of available ECG recordings, average ECGs duration in seconds, average age of patients in years, sex distribution of patients, and employed sample frequency, for each of the listed datasets.

Table 3.1: Number of recordings (# of ECGs), average duration of recordings in seconds (Avg. duration (s)), average age of patients in years (Avg. age (years)), distribution of the sex of patients (Sex (M% / F%)), and sample frequency of recordings (F_S (Hz)) for each dataset that is contained in the publicly available subset of the 2020 PhysioNet/Computing in Cardiology Challenge dataset.

Dataset	# of ECGs	Avg. duration (s)	Avg. age (years)	Sex (M%/F%)	F_S (Hz)
CPSC2018	6,877	15.9	60.2	54% / 46%	500
CPSC-Extra	3,453	15.9	63.7	54% / 46%	500
INCART	72	1800.0	56.0	54% / 46%	257
PTB	516	110.8	56.3	73% / 27%	1000
PTB-XL	21,837	10.0	59.8	52% / 48%	500
G12EC	10,344	10.0	60.5	54% / 46%	500

In Figure 3.1 we report the available ECGs for each scored label and dataset. The CA reported in the figure are listed as follows: I-degree AtrioVentricular Block (IAVB), Atrial Fibrillation (AF), Atrial FLutter (AFL), Bradycardia (Brady), Complete Right Bundle Branch Block (CRBBB), Incomplete Right Bundle Branch Block (IRBBB), Left Anterior Fascicular Block (LAnFB), Left Axis Deviation (LAD), Left Bundle Branch Block (LBBB), Low QRS Voltages (LQRSV), Non-Specific IntraVentricular Conduction Disorder (NSIVCB), Pacing Rhythm (PR), Premature Atrial Contraction (PAC), Premature Ventricular Contractions (PVC), Prolonged PR interval (LPR), Prolonged QT interval (LQT), Q wave abnormal (QAb), Right Axis Deviation (RAD), Right Bundle Branch Block (RBBB), Sinus Arrhythmia (SA), Sinus Bradycardia (SB), Normal Sinus Rhythm (NSR), Sinus Tachycardia (STach), SupraVentricular Premature Beats (SVPB), T wave Abnormal (TAb), T wave Inversion (TInv), and Ventricular Premature Beats (VPB).

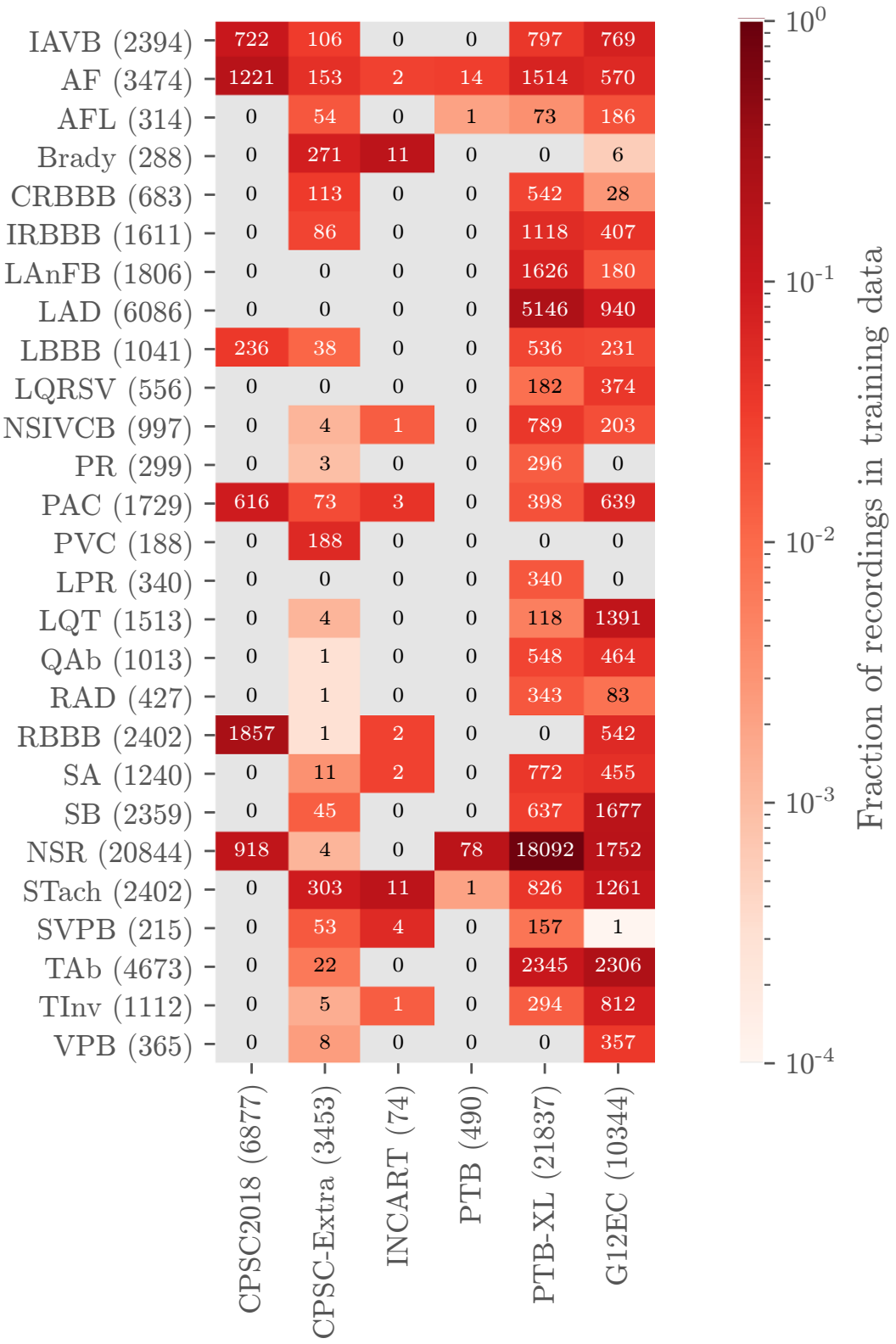


Figure 3.1: The available ECGs for each scored label, and for each of the presented datasets. The displayed colors were normalized by the total number of recordings available in each dataset. Within the parenthesis we report the total numbers of ECGs with associated a given label when merging the datasets (rows), and the total numbers of ECGs including the ones without scored labels in each dataset (columns). Credits: Perez Alday *et al.* [68], Creative Commons Attribution 4.0 International License, via medRxiv available at `https://doi.org/10.1101/2020.08.11.20172601`.

3.2.3 Preprocessing of the Electrocardiograms

ECGs were downsampled or upsampled to $500Hz$ according to their actual sampling rate and filtered with a bandpass Butterworth filter (3^{rd} order, zero phase, and pass-band: $0.67 - 30Hz$) to reduce powerline interference, baseline wandering and high frequency noise. Only the first 1 minute segment of each ECG was processed.

Beats were detected on the vector magnitude[1] (VM) signal by employing the *gqrs* algorithm [46], and beat positions were refined using the Woody algorithm applied to the VM [202]. ECG quality was assessed computing the average crosscorrelation between each QRS complex and an average QRS template. ECGs were further considered only when the signal quality was higher than 0.9 for each lead. After quality check, 4,752 signals were detected as bad quality and discarded.

Depending on the CA to detect, we processed the ECG signals differently. First, given the fact that CA altering the ECG morphology were not transient, we created an average PQRST template, *i.e.* from R peak $-260ms$ to R+$370ms$, for each lead that were concatenated afterwards. Figure 3.2 reports two examples of such concatenated vector. Second, for the rhythm-related CA, we extracted the following features from the R-R: R-R median, R-R standard deviation, R-R minimum distance, R-R maximum distance, and root mean square of successive differences of R-R. Third, for detecting PVC, we computed the maximum amplitude on the VM signal.

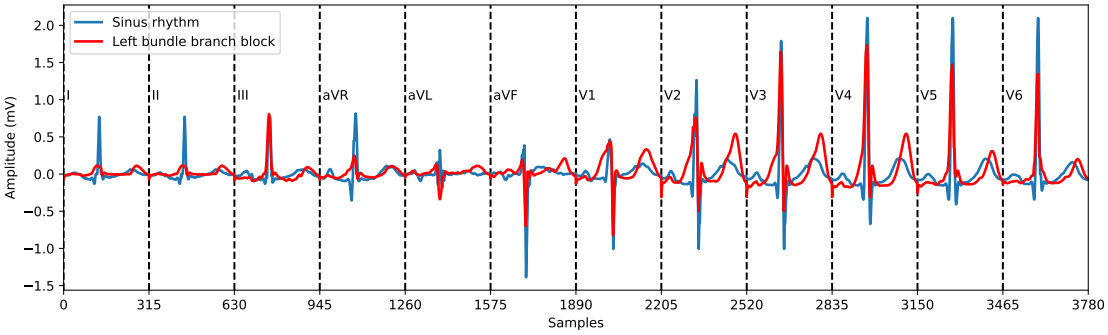


Figure 3.2: Example of average 12-lead PQRST template for normal sinus rhythm (blue line) and left bundle branch block (red line).

[1]The vector magnitude is computed as the square root of the sum of the squared ECG leads.

3.2.4 The Ensemble Model

We designed an ensemble model comprising of four CNN and a threshold-based classifier. Figure 3.3 reports the complete scheme of the ensemble model.

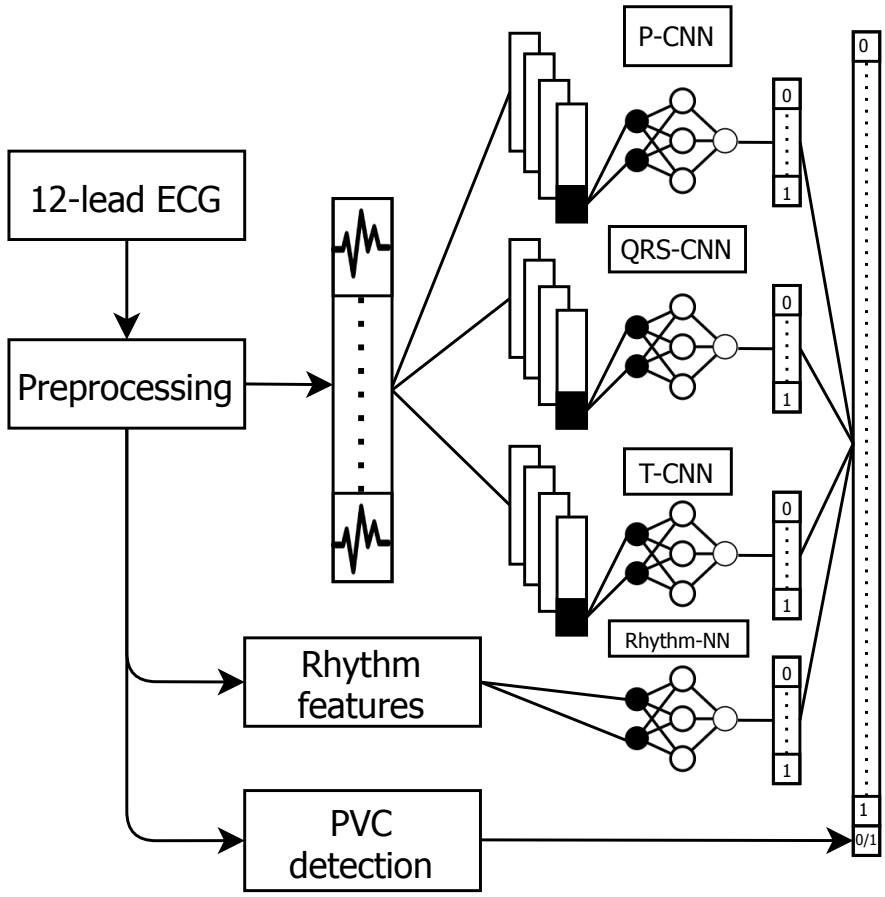


Figure 3.3: A scheme of the proposed ensemble classification model which is composed of: 1) three CNN models to classify morphology-related CA. 2) A fully connected neural network to classify rhythm-related CA. 3) A threshold-based classifier for premature ventricular beat detection. The final classification output is obtained by combining the outputs of the five classifiers.

Three CNN architectures, *i.e.* P-CNN, QRS-CNN and T-CNN, were designed to classify CA altering the morphology of the P, QRS and T segments, respectively. Each network classified within different classes:

- P-CNN classified I-AVB and LPR;
- QRS-CNN classified RBBB, IRBBB, LAnFB, LAD, LBBB, LQRSV, NSIVCB, QAb, and RAD;

- T-CNN classified LQT, TAb, and TInv.

The input features of the three CNNs were the respective concatenated P, QRS, and T average segments taken from each lead of the average beat. Specifically, P segments spanned in the range (R−260ms, R−150ms), QRS complexes were taken in the range (R−50ms, R+50ms) and T segments ranged in (R+100ms, R+370ms). Each CNN was composed of one or more convolutional layers, a dense fully connected layer and an output layer whose dimension depended on the number of classes to classify. The structure of the three CNN architectures is shown in Figure 3.4.

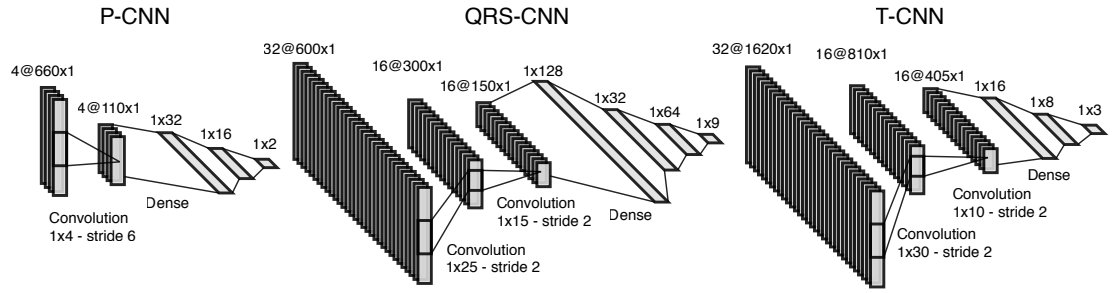


Figure 3.4: The network structure of the three CNN architectures which compose part of the the full ensemble ML model. The shapes of the convolutional kernels are reported as # of filters @ $\text{dimension}_1 \times \text{dimension}_2$.

A deep feed-forward neural network was designed to classify the CA related to irregular rhythms (hereafter, named as Rhythm−NN). The classes were AF, AFL, Brady, PR, PAC, SA, SB, and STach. The input features were those extracted from the RR series (see Section 3.2.3). The network had two hidden layers with 64 and 32 neurons, respectively, and an output layer with 8 neurons, equivalent to the number of rhythm classes.

ECG containing PVCs were classified using a threshold calibrated by means of a Receiving Operating Curve analysis performed on the maximum value of the VM signal. The optimal cut-off was selected as that one balancing the true positive and negative rates.

For all the networks, the ReLU activation function was used for the fully connected layers, and the Sigmoid activation function in the output layer [45]. No activation functions were set after the convolutional layers. Batch normalization

[155] and dropout [96] (with a rate starting at 0.1 in the first layer and with a 0.1 increase each further layer) were used in all the layers, except the last one, as regularization techniques. The Adam algorithm [203] was used as optimizer ($\epsilon = 10^{-8}$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$) and the average binary cross-entropy across classes was set as loss function [45]. The batch size was set to 64 samples.

For training the ensemble model, four datasets were built containing only the input features within to the considered subset of CA. Then, each dataset was randomly sampled with stratification using a 70/30 training/validation split. Models were trained separately for 1000 epochs on their respective training set. Metrics were computed on the validation set to assess the performance. The model submitted for the evaluation on the private test set was trained using all the available data without splitting and using the same configuration.

Given the fact that the output of the CNN architectures and the feed-forward neural network were computed by means of Sigmoid functions, resembling then the conditional probability of observing a given class, the final decision was taken by setting a 0.5 threshold for such probabilities. The output vector was obtained by concatenating all the decisions obtained by the CNN architetures, the feed-forward neural network, and the threshold-based classifier. Regarding the decision related with the detection of NSR, ECGs were classified as NSR only if no other CA was detected, *i.e.* when the concatenated output vector was equal to the zero vector.

3.2.5 Experimental Results

Given the multi-label classification problem, the confusion matrices for the four neural networks composing the ensemble model were computed on our validation sets in class-wise manner, and results are reported in Table 3.2. Positive classes contained CA specific to the neural network under evaluation, while the negative classes contained all the others. Confusion matrices were normalized by row, *i.e.* dividing by the total number of samples for each class.

We computed the recall values for all the 24 scored classes provided with the dataset on our validation sets. The three highest recall values were obtained by

Table 3.2: Confusion matrices for the networks composing the ensemble model, computed on internal validation sets. The values of each matrix were normalized by the row.

	P−CNN		QRS−CNN	
	Pred+	Pred−	Pred+	Pred−
Act+	0.71	0.29	0.87	0.13
Act−	0.12	0.88	0.19	0.81

	T−CNN		Rhythm−NN	
	Pred+	Pred−	Pred+	Pred−
Act+	0.76	0.24	0.74	0.26
Act−	0.15	0.85	0.15	0.85

the QRS-CNN for RBBB and LBBB (0.93 and 0.85, respectively) and by P-CNN for I-AVB (0.88). The worst values were achieved by the Rhythm-CNN for Pacing (0.71) and Flutter (0.74), and for the normal ECG detection (0.74). The area under the ROC curve for the PVC detection was 0.82, obtaining true positive and negative rates of 0.72. The identified threshold was $1.44mV$.

The challenge scoring system made use of a metric depending on the recognition performance of each class in a weighted manner. The employed misclassification cost was defined for each of the considered CA by expert physicians in Perez Alday *et al.* [68]. Organizers made available the scoring system: we obtained a score of 0.241 on the private validation set and a score of −0.179 on the private test set.

3.2.6 Discussion

The ensemble model reached intermediate classification performance. The QRS-CNN was the best among the four NN models and it reached the top highest recall values (up to 0.93 for RBBB) computed over all the 24 classes. The other three networks and the PVC detector showed moderate performance, reporting the worst recall values. We noticed that the worst performance were still correlated with classes having a low number of samples. For instance, the P-CNN model was trained using only the available 340 samples for prolonged PR, achieving one of the lowest recall values (0.74).

Several are the improvements that can be implemented. First, the Rhythm-NN and PVC detector can be substituted with more efficient models. In fact, Hannun

et al. [67] and Zhou *et al.* [204] recently demonstrated that DNN can achieve high recall values for both rhythm and PVC detection. Second, the low recall values of the T-CNN might be due to the preprocessing step implemented. Indeed, the average PQRST template did not account for changes in the heart rate within the considered 1-minute segment, while it is well known the heart-rate dependency of the T wave morphology and duration. R-R binning can be used to improve this aspect instead of averaging beats within the entire segment [205]. Third, the recall value for normal ECG detection was among the lowest ones. The detection by elimination, *i.e.* when the final output was the zero vector, was sensitive to misclassification of any of the other classes. For example, if misclassifications were statistical independent between the classes and the error rate was just random at 1% (but we are still far from this value for many classes), the misclassification of normal ECG would be approximately 23%, leading to an extremely high false negative rate. A possible solution might be designing and adding another CNN in the ensemble model, whose input is the average PQRST template and several rhythm-related features, capable of recognizing normal sinus rhythm ECGs.

Differently from previous studies on DNN where DL architectures were often borrowed from other domains, we designed a hybrid approach merging average-template-based algorithms, known to be effective, with the state-of-the-art for classification in deep learning, by using an ensemble model. The approach seemed suitable to deal efficiently with the challenging multi-class problem of ECG classification, and the limited sample size available. Future works are towards the testing of further DL algorithms. For instance, we expect that recurrent neural networks would lead to better performance in the classification of rhythm-related CA with respect to the employed feed-forward neural network, since they already showed to be effective for their classification [48, 49, 55].

3.3 Classification of 12-lead Electrocardiograms with Different Lead Systems Using Automated Machine Learning

3.3.1 Introduction

In the context of Bodini *et al.* [110] we took part to the 2021 PhysioNet/Computing in Cardiology challenge [69], which asked participants to perform the automatic classification of 30 CA from both 12-lead ECGs and reduced-lead settings. We investigated on the feasibility of applying AutoML approaches to build ECG classifiers.

Standard ECG preprocessing was applied beforehand to the ECG (filtering and resampling). Three different AutoML frameworks were executed on the 88,000+ ECGs made available by the challenge organizers. The optimal combination of preprocessing and ML algorithms were found by the AutoML frameworks. We finally assessed the frameworks' classification performance, the effect of the number of employed leads, and the effect of extending the frameworks training time.

The classifiers proposed by our team "BiSP_Lab" received scores of 0.30, 0.29, 0.28, 0.26, 0.23 (ranked 27th, 29th, 28th, 29th, 28th out of 39 teams) for the 12-lead, 6-lead, 4-lead, 3-lead, and 2-lead versions of the hidden test set with the challenge evaluation metric. The AutoML frameworks showed comparable performance, and the worst score was obtained on the 12-lead system, while the best on the 6-lead one. Significantly extending the training time seemed to not improve the test score. The obtained results showed that AutoML frameworks obtained promising performance on the private test set, suggesting their potential for classification of CA.

3.3.2 The 2021 PhysioNet/Computing in Cardiology Challenge Dataset

The dataset made available for the challenge was composed of 12-lead, 6-lead, 4-lead, 3-lead, and 2-lead ECGs, labeled with one or more CA, among 133 possible ones [69]. The provided dataset by the organizers was created by extending the 2020 Physionet/Computing in Cardiology challenge dataset [68], which we presented in Section 3.2.2. Regarding the publicly available dataset, it was integrated with

respect to the previous year challenge by merging two further sources: 1) The Chapman-Shaoxing database, which contains 10,247 recordings from the Chapman University (Orange, California, United States) and Shaoxing People's Hospital (Shaoxing, China) [63]. We yet presented the latter database in Section 1.3. 2) The Ningbo dataset which contains 34,905 recordings from Ningbo First Hospital (Ningbo, China) [206]. A total number of 88,253 ECG signals was made publicly available. Like the previous year challenge, the performance of the submitted classifiers were assessed using an expert-based scoring metric provided by the challenge organizers which assessed the classifiers' performance only relying on a subset of 30 selected CA [68]. A hidden validation and test sets, respectively composed of 6,630 and 36,272 recordings, handled by the challenge organizers, were used for evaluating the proposed algorithms. A maximum of 72h was allowed for training time and 24h for testing.

3.3.3 Preprocessing of the Electrocardiograms

Similarly to the previous year challenge [109], we only relied on the public dataset provided for the challenge, and on signals labeled with the CA considered in the challenge scoring metric defined by the organizers. Furthermore, we merged the classes which the committee scored as the same diagnosis. Standard ECG preprocessing was applied beforehand to raw ECGs, including filtering and resampling. ECGs were downsampled or upsampled to $125Hz$ according to their actual sampling rate and filtered with a bandpass Butterworth filter (3^{rd} order, zero phase, and pass-band: $0.67 - 30Hz$) to reduce powerline interference, baseline wandering and high frequency noise. For each lead system, only the first $10s$ of ECG were considered. In case the length was inferior to $10s$, zero padding was performed. The available ECGs were randomly split into training and validation sets with 70/30 ratio for each lead system, with stratification (*i.e.* the class distribution of the training set matched the one of the validation set).

3.3.4 The Automated Machine Learning Frameworks

We adopted three different AutoML frameworks to build the required ECG classifiers, *i.e.* auto-sklearn, AutoKeras, and the Tree-Based Pipeline Optimization Tool (TPOT). The AutoML frameworks are capable of automatically selecting the optimal ML pipeline to solve the problem at hand. With the term "pipeline", we hereinafter refer to the process of rebalancing the distribution of classes, feature engineering, dimensionality reduction, and training of the selected ML/DL classifier, along with its parameters and hyperparameters [207, 208].

Class rebalance methods applied by AutoML frameworks address the classes imbalance problem that we briefly introduced in Section 1.4. Datasets with imbalanced classes are frequent in several real-world applications, including ECG classification where the classes distribution is often not balanced since it is sometimes hard to collect ECGs related to certain CA for several circumstances (*e.g.* in the case of rare CA, and unavailability of patients showing certain CA at acquisition time) [99]. Several methods were introduced to address such problem and they are usually divided into data level rebalance methods, and algorithm level rebalance methods [99, 106]. The data level rebalance methods change the distribution of the dataset by oversampling or undersampling [102, 106]. The algorithm level rebalance methods adjust the underlying ML algorithm without altering the original distribution of training data. Examples of the latter methods include thresholding, cost-sensitive learning, and one-class classification [106]. Threshold methods are applied once the ML model is learned (hence, during the test phase) to adjust its decision threshold by changing the output class probabilities. Cost-sensitive learning assigns different costs to the misclassification errors for training samples from different classes [107], and it can be implemented in several ways depending on the underlying ML or DL algorithm. For instance, if considering DNN, a possible approach is to train a DNN to minimize a specifically defined misclassification cost, instead of the commonly employed loss functions such as cross-entropy [107]. One-class classification is usually called novelty or anomaly detection in the context of DNN [168, 209]: to handle the classification issues related to the minority lass, a

DNN could learn to recognize the samples associated to the majority class, rather than discriminating between the two classes [168, 209].

The AutoML frameworks can potentially generate a huge amount of features. The higher is the number of considered features, the higher will be the computational cost for obtaining them, and the impact on the training time of ML algorithms [210]. Furthermore, it happens that several computed features may be not relevant to distinguish within the classes of the problem at hand, or they may be correlated, thus leading in both the cases to a huge number of irrelevant inputs. The presence of correlated features negatively affect the performance of several ML models in different ways and to varying extents [211, 212]. As a consequence, it is often critical to decrease the dimensionality of the computed features by discarding not discriminative, and correlated features for both improving computational efficiency and classification performance [211, 212]. Typically, the process of reducing the dimensionality of the involved features is referred in the literature as *dimensionality reduction* [211, 212], and it is achieved prior to the training step of ML algorithms through feature selection or feature extraction methods [48, 114]. Feature selection methods attempt to determine a subset of the initial feature space by evaluating the impact of the computed features on the final classification performance. The selected subset should be capable of both properly representing the input data, and of providing adequate or even improved classification performance, while allowing to reduce the computational training time of the employed ML algorithm [211, 212]. The available techniques to perform feature selection can be mainly divided in filter methods, wrapper methods, and embedded feature selection methods [211, 212]. Filter-based feature selection methods employ a certain metric to identify and remove less relevant features. The selecting procedure of the less useful features is separated from the training step of the employed ML algorithm. In Wrapper-based feature selection methods the process of feature selection is executed during the training phase, and the model's classification performance metric is employed as a feature selection criterion. The Sequential Forward Selection (SFS) and Sequential Backward Selection (SBS) methods [213] are within the most employed wrapper

methods [211, 212]. In embedded feature selection approaches, the selection of the optimal features subset is implemented into the classifier design process, thus it is conducted automatically during training phases. Regularization constraints in the form of L_p norms were applied to several ML classification algorithms as embedded feature selection methods [211, 212].

The feature selection approaches discard the not selected features relying on a certain strategy. On the other hand, the process of feature extraction maps the full set of features on a space with lower dimensionality, by keeping the most relevant information of the original space to the maximum possible extent [48, 114]. We notice that the aim of both feature selection and feature extraction methods is to reduce the dimension of the original feature space, hence these two terminologies are often used interchangeably by researchers, but they are not equivalent [114]. Among the most common feature extraction techniques we may find Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Independent Component Analysis (ICA), and several other ones [48, 114].

The auto-sklearn framework is an AutoML system based on the Python scikit-learn library [214]. It relies on 15 ML classifiers, 4 data preprocessing methods, and 14 feature preprocessing methods, which give rise to a high-dimensional hypothesis space. Onto such space, auto-sklearn defines a Combined Algorithm Selection and Hyperparameter optimization (CASH) problem and it relies on Bayesian Optimization to optimize such problem for discovering a top-performing ML pipeline.

The AutoKeras framework is an AutoML tool specific for DL architectures, based on the Python Keras library [215]. It exploits the concept of network morphism, which retains the functionality of a DL network while changing its underlying architecture. Bayesian optimization is leveraged by AutoKeras to guide the network morphism in searching of the optimal DL architecture for the considered problem and dataset. To efficiently explore the search space, the authors of the AutoKeras framework developed a custom neural network kernel along with a tree-structured optimization algorithm.

The TPOT framework automatically constructs and optimizes ML pipelines relying on the well-known evolutionary computation technique of genetic programming (GP) [216]. At the beginning of every TPOT run, a fixed number of pipelines is generated to constitute what is usually called in GP as population. GP is used to evolve the set of pipelines that acted on the dataset, and a portion of those is retained relying on their classification performance. The top-performing pipeline is retained when TPOT reaches convergence or after a user-defined number of runs.

3.3.5 Experiments on the Frameworks

The AutoML frameworks were trained on the available dataset with the aim of 1) comparing the performance among the three considered frameworks; 2) assessing the effect of the number of employed leads on the final classification performance; 3) assessing the effect of extending the training time at disposal of the AutoML frameworks.

The input features were set as the reduced $10s$ ECGs for each lead system, and the respective validation sets were employed by the frameworks to select the optimal ML pipeline. By default, AutoML frameworks use the validation loss of the employed ML/DL algorithm as a score for selecting the best pipeline. For each AutoML framework, we followed a cost-sensitive learning approach by setting the challenge score defined by organizers as scoring function to measure the performance of the created pipelines. Each AutoML instance comes with a parenthesized name to easily refer to it.

Auto-sklearn was tested by setting $2.5h$ of training time for each lead system (auto-sklearn #1), and setting a proportional training time to the number of leads of ($2.5h \times$ #leads) for each lead configuration (auto-sklearn #2). The whole set of classifiers, feature preprocessing methods, and data preprocessing methods was considered.

AutoKeras was tested relying on a training time of ($2.5h \times$ #leads) for each lead configuration and using the full set of pipeline elements at disposal (AutoKeras #1). Next, the hypothesis space was reduced to consider only DL architectures composed

of Convolutional, Dense, ResNet, and Xception layers, and by mantaining the same training time of the previous configuration (AutoKeras #2).

TPOT was tested under two configurations with a training time of $2.5h \times$ #leads. The TPOT Default configuration was used to search over a broad range of pipeline elements, where some of them may take a long time to run, especially on large datasets (TPOT #1). Then, the TPOT Light configuration was tested, in which TPOT searched over a restricted range of simple and fast-running pipeline element to find quick and simple ML pipelines (TPOT #2).

To assess the effect of extending the training time at disposal of the AutoML frameworks, as a final test we considered only the 3-lead system and we trained each AutoML framework for $70h$ onto such system (while we used pre-trained ML models for the remaining lead systems in the instance submission phase). For each AutoML system, the settings of the top performance instance were selected.

3.3.6 Experimental Results

To compare the performance among the considered AutoML frameworks, we computed in the Figure 3.5a the cumulative sum of challenge scores obtained by the AutoML instances on each lead configuration, on the hidden validation set. The sub-bars report the score obtained by an AutoML instance on a specific lead system. To assess the overall variability of all the AutoML instances over lead systems, we reported in the Figure 3.5b the box-plots of challenge scores computed over each lead system on the hidden validation set.

To quantify the effect of the number of employed leads on the final classification performance, similarly to Figure 3.5a, we computed in the Figure 3.5c the cumulative sum of the challenge scores obtained over each lead system by the AutoML instances on the hidden validation set. For each lead system, the sub-bars report the score obtained by an AutoML instance. To assess the variability of a specific AutoML instance over lead systems, we reported in the Figure 3.5d the box-plots of challenge scores computed over each lead system on the hidden validation set.

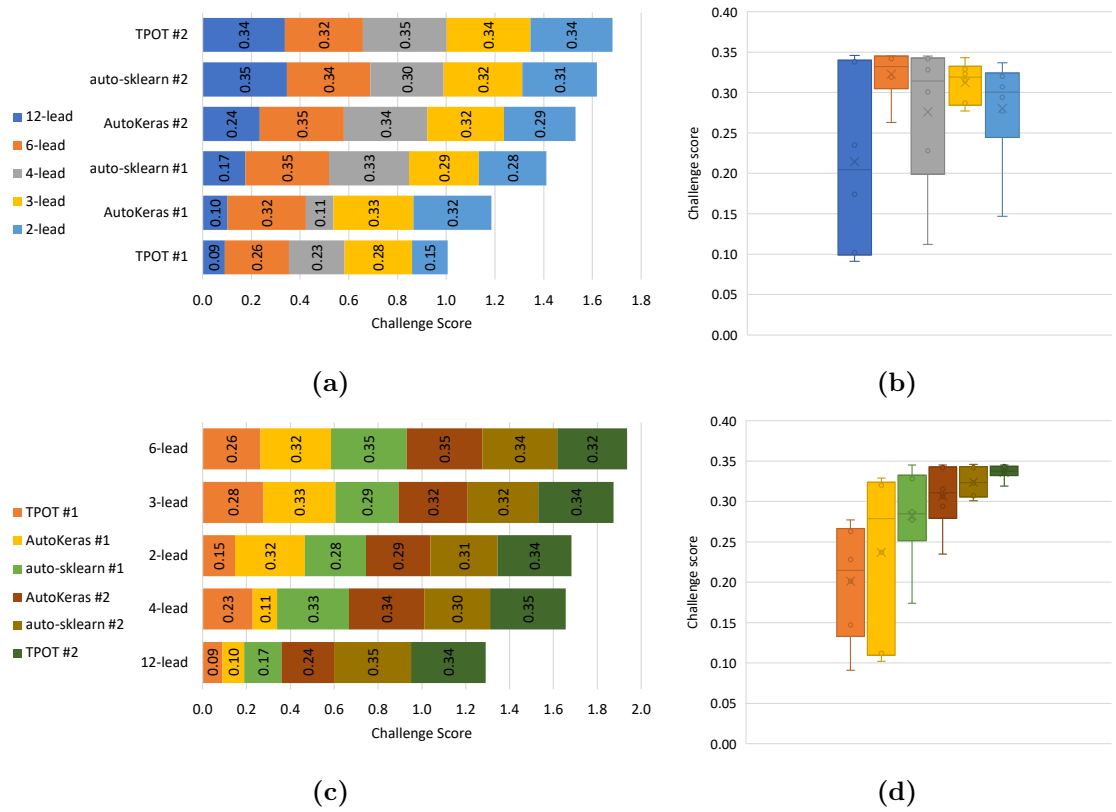


Figure 3.5: (a) The cumulative sum of challenge scores of AutoML instances on each lead system. (b) The box-plots of the challenge scores computed over each lead system. (c) The cumulative sum of challenge scores obtained over each lead system by the AutoML instances. (d) The box-plots of challenge scores computed over each AutoML instance.

After training for $70h$ the top-performing instances on the 3-lead system for each AutoML framework, *i.e.* auto-sklearn #2, AutoKeras #1, and TPOT #2, we respectively obtained 0.32, 0.33, and 0.35 challenge scores on the hidden validation set.

Table 3.3 reports the challenge score on the hidden validation and test sets, achieved by the final selected entry (auto-sklearn #2).

Leads	Validation	Test	Ranking
12	0.35	0.30	27th
6	0.34	0.29	29th
4	0.30	0.28	28th
3	0.32	0.26	29th
2	0.31	0.23	28th

Table 3.3: Challenge scores for our final selected entry (auto-sklearn #2) on the hidden validation and test sets, as well as the ranking on the hidden test set.

3.3.7 Discussion

As shown in Figure 3.5a, TPOT #2 was the best among the six instances in terms of cumulated score, and it reached the highest score values in three out of five lead systems (up to 0.35 with 4-leads). The worst performance were provided by TPOT #1, since it reached the lowest challenge score values in four out of five lead systems. The Figure 3.5d shows that TPOT #2 and TPOT #1 were the ones with the highest and lowest median challenge score value computed across lead systems, respectively of 0.34 and 0.21. The instance TPOT #2 showed the lowest interquartile range (IQR) of 0.01, while the highest IQR of 0.14 was reached by TPOT #1. The highest IQR obtained by TPOT #1, associated with the lowest median score value, suggests that in this case the used AutoML configuration may be weak in classifying CA, since it searched into a limited hypothesis space.

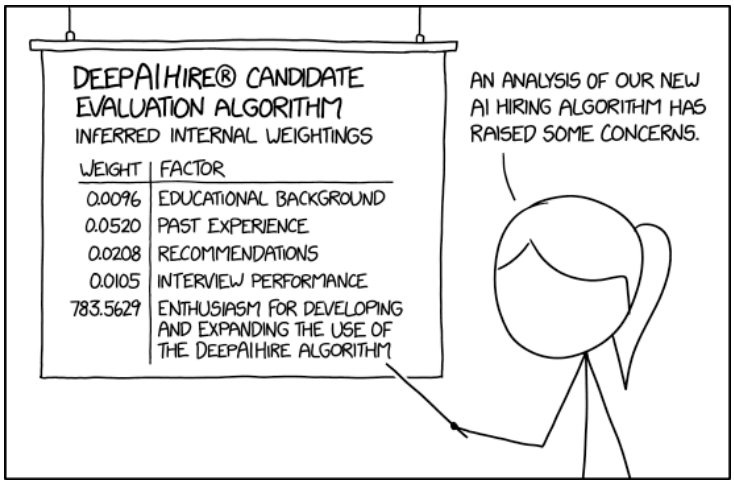
The Figure 3.5c shows that the 12-lead configuration is the one where instances obtained the lowest performance. On the other hand, the 6-lead system showed the highest cumulated score. The Figure 3.5b shows that the 6-lead and the 12-lead systems were the ones that showed respectively the highest and lowest median score value computed across AutoML instances, respectively of 0.33 and 0.20. The 6-lead system obtained the lowest IQR of 0.04, while the highest IQR of 0.24 was obtained by the 12-lead system. The results on the 3-lead are comparable the to 6-lead system, as it is reasonable to expect since the second system is a linear combination of former [18]. The highest IQR obtained by the 12-lead system, associated with the lowest median score value, suggests that in this case AutoML frameworks might need further training time to match the performance obtained in the case of less numerous lead systems.

The experimented AutoML frameworks obtained intermediate classification performance with respect to other teams. Since the class distribution of the available dataset was not balanced, a cost-sensitive learning approach was leveraged to face the class imbalance problem: we executed the AutoML frameworks to find optimal pipelines relying on the challenge score function, instead of the standard loss functions of the employed ML/DL algorithms. The misclassification costs

defined for CA by expert physicians helped in learning the few represented class, by considering their misclassification cost within the knowledge domain.

Further aspects of AutoML frameworks may be explored in the future. A wide number of AutoML tools is arising in the recent literature and more AutoML frameworks may be tested to address ECG classification, even if recent works suggest that their performance is relatively similar [207]. Next, even if the impact of increasing the training time did not significantly improve the performance in the case of the 3-lead system (not more than 0.03 of the challenge score), a systematic assessment of performance against training time may be investigated even for other lead configurations. A full analysis was not performed due to the limited number of instance submissions available, that was 10, and limited time for training ($72h$). However, the missed improvement in performance may be in line with results of recent works, which showed that most of AutoML frameworks tend to converge to similar performance in a few hundreds of iterations [207].

Differently from previous works where ML algorithms where often applied without a deep exploration of ML pipelines, and designs of DL architectures were inspired from other domains, we tested AutoML approaches to manage the proper choice of the optimal ML pipeline, and at the same time to face the class imbalance problem with the aim of cost-sensitive learning. The approach seemed suitable to deal efficiently with the challenging problem of ECG classification and the unbalanced dataset available.



4

Explainability of Machine Learning Algorithms for Classification of Cardiac Abnormalities

Contents

4.1 Introduction

In the present Chapter 4 we introduce our research works which were focused on explainability of ML algorithms in the classification of CA from ECG signals.

In Section 4.2 we will present our research work where we relied on the Local Interpretable Model-agnostic Explanations (LIME) [183] explainability algorithm to highlight which ECG leads were the most relevant for a random forest algorithm in the classification of several kinds of ST-Elevation Myocardial Infarction, depending on their anatomical localization [98]. In the work we showed how to overcome the overfitting problem caused by an inherent bias that we found in the employed dataset. Furthermore, differently from the majority of other works presented in Section 1.4 which addressed the problem of explainability in ECG classification, we properly framed the explanations in the domain knowledge of electrocardiography by designing a custom metric which allowed us to highlight the importance of each lead in the final classification output, and we averaged the explanations over all the training ECGs.

In Section 4.3 we will present our research work where we designed two explainability frameworks relying on two model-specific, *post-hoc*, and local explainability algorithms for Convolutional Neural Network (CNN) architectures to explain which ECG waves were the most relevant in the classification of CA for a state-of-the-art CNN [111]. The introduced frameworks could be useful from the perspective of the ML expert, since they allow to inspect if a trained CNN correctly relies on the same ECG segments which physicians are expected to look at in the usual diagnosis of CA. From the perspective of physicians, the proposed frameworks allow them to understand whether the classification output provided by the CNN relied on their domain knowledge, by highlighting the expected ECG waves assessed during diagnosis, and thus fostering the trust in the employment of DL. To the best of our knowledge, with the mentioned work we were the first, at the same time with

Zhang *et al.* [94], to systematically evaluate the performance of a CNN against the domain knowledge of ECG interpretation on a wide set of 27 different CA.

4.2 Explainability of Machine Learning Algorithms in the Classification of ST-Elevation Myocardial Infarction

4.2.1 Introduction

ST-Elevation Myocardial Infarction (STEMI) is a common cardiovascular disease that is caused by the occlusion of one or more coronary arteries [112]. A severe occlusion of coronary arteries leads to ischemia, *i.e.* a significant reduction of blood flow to the myocardium. In the presence of a prolonged ischemia, the oxygen supply becomes insufficient and it occurs a necrosis of the surrounding cardiac tissues, causing an acute heart attack [112]. STEMI is one of the leading causes of death for humans: as we saw in Section 1.1, according to World Health Organization an estimate of 7.3 million people die annually due to cardiovascular diseases, representing 31% of all global deaths, and of these deaths roughly the half are caused by heart attacks [2]. Therefore, an accurate and early detection of STEMI is fundamental to increase life expectancy and to improve life quality.

The ECG analysis is a crucial step in the diagnostic triage of patients with suspected STEMI. Physicians acquire the 12-leads ECG and usually assess the ST-segment Elevation (STE), which is the most commonly linked marker to coronary occlusion. The STE persists on the ECG for several weeks after an acute infarct [19]. The location of STEMI can be derived recalling the anatomical area that pertains to the involved leads which present a STE [112]. Furthermore, it must be noticed that the diagnosis of STEMI must be confirmed with differential diagnosis (advanced cardiac ischemia usually may cause chest pain [217]), and with the presence of specific biomarkers (for instance, high cardiac Troponin values [112]). However, the ECG is regarded as the most effective tool for the prompt diagnosis of STEMI, as it is inexpensive, quickly performed, and rapidly available [19].

Several years of training are required for a physician to become expert in ECG-based CA interpretation, and even for expert physicians the manual interpretation of multiple ECG traces is a time consuming task [20]. Furthermore, several studies highlighted that many regions of the world have a low doctor-patient ratio that makes the access to health care difficult [29, 30, 35]. As we saw in Section 1.2, being the ECG one the most effective tool for the prompt diagnosis of STEMI, to complement the role of physicians computer-aided diagnosis programs have been widely developed and have been gaining high attention worldwide [21, 24]. Several computerized diagnosis programs are nowadays used by physicians with high consensus [24], for instance the University of Glasgow (Uni-G) ECG analysis program [28].

Focusing on STEMI, researchers proposed several ECG classification systems based on ML algorithms [48, 49, 55, 114]. For instance, standard ML algorithms were leveraged relying on features extracted according to the medical expertise [218, 219]. Furthermore, since in other fields avoiding the step of feature engineering provided remarkable results, black-box DL algorithms that automatically learn useful features from the ECG have been recently introduced [220–222].

Despite latest ML and DL models for ECG classification reached remarkable classification performance, they often lack of explainability. Thus, we investigated on an explainability method capable of providing explanations of their classification outputs. We specifically employed the Local Interpretable Model-agnostic Explanations (LIME), a model-agnostic, *post-hoc*, and local explainability method which explains the classification outputs of a ML classifier revealing which part of the input most contributed to the classification [183]. LIME let us to understand whether ML classifiers consider significant features for STEMI, and hence to correct their functioning if it is the case, thus likely increasing the performance and fostering the trust in their outcomes. In order to validate the explanations provided by LIME, we compared them with the anatomical position of the infarct, known as part of the diagnostic report of the patient.

4.2.2 The Physikalisch-Technische Bundesanstalt Dataset

ECGs were taken from the Physikalisch-Technische Bundesanstalt (PTB) dataset [46, 65, 66]. The ECGs available in the dataset were collected from healthy subjects and patients with several heart diseases by Prof. Michael Oeff, at the Department of Cardiology of University Clinic Benjamin Franklin (Berlin, Germany).

The database contains 549 acquisitions from 290 subjects (aged 17 to 87, mean 57.2; 81 women). Each subject is represented by one to five records. Each record includes 15 simultaneously measured signals: the conventional 12 leads (I, II, III, aVR, aVL, aVF, V1, V2, V3, V4, V5, V6) together with the 3 Frank leads ECGs (Vx, Vy, Vz). ECGs were sampled at $1kHz$, 16 bit resolution, and have variable length (the typical duration is around two minutes). We considered only the 12 standard leads. For each ECG a detailed clinical summary is available, including age, gender, diagnosis, and where applicable, data on medical history, medication and interventions, coronary artery pathology, ventriculography, echocardiography, hemodynamics, and anatomical position of eventual STEMI. The clinical summary is not reported for 22 subjects.

The PTB database contained 368 traces for 148 STEMI patients and 80 traces for 52 Healthy Control (HC) subjects. For STEMI, we selected only the 341 traces whose anatomical infarct location was annotated.

4.2.3 Preprocessing of the Electrocardiograms

Selected ECGs were filtered with a bandpass Butterworth filter (3^{rd} order, zero phase, and pass-band: $0.67 - 30Hz$) to reduce powerline interference, baseline wandering and high frequency noise. The baseline of ECGs was adjusted: for each lead, the mode of the ECG's samples distribution (with a bin size of $75\mu V$) was computed. Then, the average of the samples belonging to the modal bin was subtracted from the signal [223].

Beats were detected on the vector magnitude signal using the *gqrs* algorithm [46]. Beat positions were aligned on the R peak using the Woody algorithm applied to the vector magnitude [202]. Quality of signals was assessed computing the mean

crosscorrelation with an average QRS template. An ECG trace was considered of good quality when such crosscorrelation was higher than 0.9 for each lead. After quality assessment, we obtained 44 HC traces, and for STEMI: 18 anterior, 15 antero-lateral, 34 antero-septal, 54 inferior, and 29 infero-lateral infarct traces. Other infarct locations were not considered since less than 10 traces were of good quality.

For each ECG, the average beat was computed for any lead. Then, two configurations were considered. First, we concatenated the average QRST segment of each lead in a single vector. The considered QRST segment spanned from $50ms$ before the R peak to $150ms$ after it, obtaining a feature vector of 2400 elements. Second, we concatenated the average ST segments only. Specifically, we considered segments from $50ms$ after the R peak up to $150ms$ after it, with a resulting feature vector of 1200 elements. We used the concatenation of average beats, and the concatenation of average ST-segments as features.

4.2.4 Training of the Random Forest Classifier

We considered the Random Forest (RF) algorithm for our proof of concept. A different RF was trained for each of the two feature vectors (the concatenations of average beats or of average ST-segments) and for each of five specific infarct positions. The binary classification approach distinguished HC from STEMI subjects. For each RF, a dataset with features from HC and STEMI subjects was built. Then, a 70/30 training/test split was sampled with stratification (*i.e.* the same proportion of classes was preserved).

The hyperparameters of the models (*i.e.* number of estimators, maximum number of leafs, maximum depth, minimum number of samples required to split nodes, and minimum number of samples required to be at a leaf) were tuned relying on a 10-fold cross validation applied to the training set. Specifically, we performed a random search by uniformly sampling 10^3 combinations in the range from 1 to 50, with a step of 10, for each parameter. In addition, Gini and Shannon entropy measures were tested as splitting criterion. The combination of hyperparameters that maximized

the validation accuracy was then retained for the final training of the RF on the entire training set. Accuracy, precision and recall were finally evaluated on the test set.

4.2.5 Explaining the Random Forest with LIME

We employed LIME, which was introduced in the details in Section 2.2.5, to explain the classification outcomes of the five RF models. Here we briefly recall that LIME is a model-agnostic, *post-hoc*, and local surrogate explanation model, *i.e.* it approximates the classification output of an instance by using a simpler linear model. The simplified model is fitted on an artificial dataset created by probing the considered black-box model *locally* on the considered instance. LIME defines the explanation model as

$$\text{explanation}(\boldsymbol{x}) = \underset{g \in G}{\operatorname{argmin}}\, \mathcal{L}(f, g, \pi), \tag{4.1}$$

where $\boldsymbol{x}$ is the instance, g is a model within the family of possible explanation models G, and $\mathcal{L}$ is the mean square error loss function, which measures how close the explanation is to the classification output of the original model f. The simpler model g is fitted by minimizing the loss $\mathcal{L}$ using an artificial dataset created by sampling in a neighbourhood of the instance $\boldsymbol{x}$. A kernel function π defines the weight of each instance of the artificial dataset based on the distance with $\boldsymbol{x}$ (higher weights are associated to lower distances).

For each of the trained RF f, we ran LIME as follows: given an instance $\boldsymbol{x}$ belonging to the training set used to train f, we generated an artificial dataset by adding to $\boldsymbol{x}$ a white Gaussian noise, with zero mean and a standard deviation of $0.5mV$, to obtain $\lfloor 10^3/\text{training set size} \rfloor$ "artificial" samples ($\lfloor \cdot \rfloor$ is the floor function). Such artificial samples were weighted according to their distance to the instance $\boldsymbol{x}$ using an isotropic Gaussian kernel π with 0.5 width. A linear model g was trained on the artificial training set with a loss function $\mathcal{L}$ defined as kernel weighted least square with L_1 norm penalizer (Lasso). The λ parameter of the Lasso method was set to 10^{-4}. We repeated the procedure for each sample of the training set.

At the end of the procedure, a Lasso weight was available for each ECG sample in the feature vector. A large weight indicated high relevance of that sample for the classification of that subject. In order to also have a Relevance measure (RV) for each lead, we computed the sum of the absolute value of the weights belonging to that lead, and normalized these 12 values with their sum. Finally, the average RV across the training set instances was computed.

4.2.6 Experimental Results

In Table 4.1, we report the accuracy, precision, and recall quantified on the test set for the considered five RF models and the two feature vectors. All metrics ranged from 0.77 to 0.92, hinting to a robust training of the RFs.

Table 4.1: Values of accuracy (Acc.), precision (Prec.), and recall (Rec.) for each infarct location: inputs are average QRST template (top), and average ST segment (bottom). The three highest RV measures are reported along with their respective lead. The table entries were colored in green or red color respectively in the case they refer to leads which are, or are not, anatomically related to the considered infarct [112]. The following acronyms were used: Anterior (AMI), Antero-lateral (ALMI), Antero-septal (ASMI), Inferior (IMI), Infero-lateral (ILMI).

Average QRST template

	Acc.	Prec.	Rec.	1st lead / RV	2nd lead / RV	3rd lead / RV
AMI	0.85	0.89	0.84	V1 / 0.24	V2 / 0.14	V4 / 0.14
ALMI	0.84	0.81	0.77	V1 / 0.31	I / 0.29	V2 / 0.09
ASMI	0.92	0.89	0.90	I / 0.22	aVF / 0.22	V1 / 0.13
IMI	0.88	0.87	0.85	II / 0.14	V1 / 0.11	V2 / 0.11
ILMI	0.89	0.88	0.79	I / 0.19	II / 0.18	V1 / 0.15

Average ST segment

	Acc.	Prec.	Rec.	1st lead / RV	2nd lead / RV	3rd lead / RV
AMI	0.91	0.82	0.81	V1 / 0.29	V2 / 0.25	V3 / 0.17
ALMI	0.89	0.83	0.79	I / 0.19	V1 / 0.17	V2 / 0.12
ASMI	0.86	0.80	0.90	V3 / 0.29	V1 / 0.21	V2 / 0.14
IMI	0.87	0.81	0.82	II / 0.44	aVF / 0.17	III / 0.11
ILMI	0.85	0.82	0.78	I / 0.28	V1 / 0.24	II / 0.09

Regarding LIME explanations and the relevance measure RV, we noticed that: 1) In the case features are average ST-segments, the highest RV values refer to the leads that anatomically pertain to the considered infarcts in 4 times out of 5.

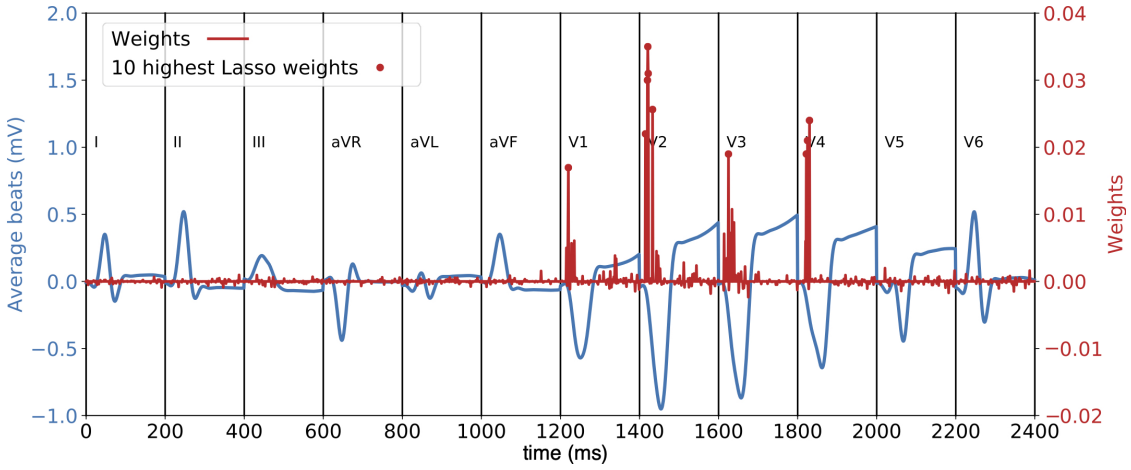
Furthermore, only 6 of the computed RV values are referred to leads which are not anatomically related to the considered infarct. 2) In the case features are average beats, the highest RV values mostly refer to leads that are not anatomically linked to the considered infarcts. Furthermore, only 5 of the computed RV values are referred to leads which are anatomically related to the considered infarct.

4.2.7 Discussion

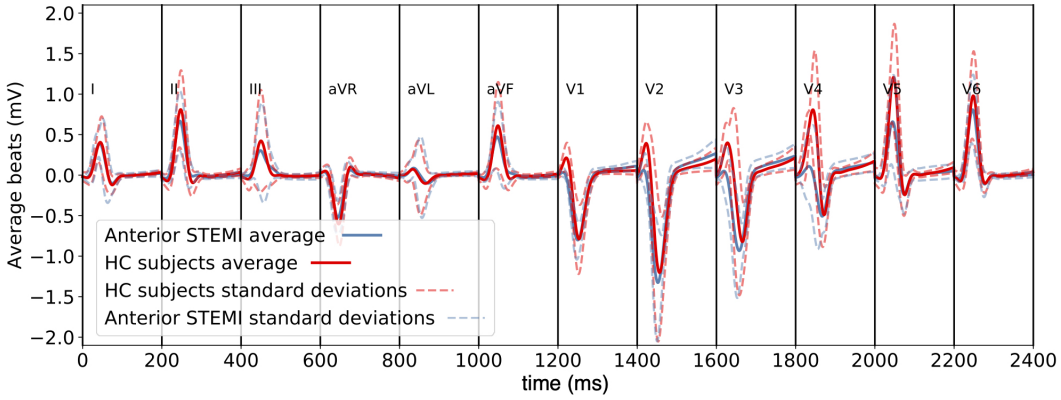
Even if the latest ECG-based STEMI ML classification algorithms usually take raw, or almost raw, ECGs as input, we performed a preprocessing phase and computed an average template representation. This choice relies on the fact that it has been observed that ST-segment elevation is the main ECG marker for STEMI [19, 112]. STE persists over the time on the ECG even for several weeks after an acute infarct, which is the case for the the patients represented in the PTB database. This procedure preserved the STE marker, reduced the noise, and proved to be efficient in terms of performance (as observed in Table 4.1). Furthermore, we followed the recommendation of the International Guidelines for myocardial infarction identification [112] by using the standard twelve lead ECG for a proper STEMI diagnosis and anatomical localization, despite several ML methods applied in this context did not rely on this standard setting [221].

While the two considered average template representations reached comparable classification performance, our analysis showed that the RF models which employed the QRST average template often relied on leads which were not anatomically related to the considered infarct (as suggested by RV in Table 4.1). On the contrary, in the case it was employed the ST average template, the RF models relied significantly more on relevant leads prescribed by the international guidelines [112]. In the case it was employed the QRST average template, LIME showed that the ECG samples mostly relevant for the classification were located on the QRS complex (Figure 4.1a), rather than on the ST segments as recommended by the international guidelines [112]. This result might be explained by observing the high variability of the QRS complex between HC and STEMI (V1, V2 and V3 in Figure 4.1b),

and implicitly suggests a low inter-subject variability in the PTB dataset. We think that this effect might be due to the age difference between the HC subjects and STEMI patients (HC: 53 ± 17 vs STEMI: 67 ± 14), as QRS narrows while ageing [224]. Thus, LIME hinted that the QRST based RF might be unreliable when used in real scenarios, since it tends to overfit on the QRS surroundings instead of the ST-segment, despite the high validation accuracy achieved. By properly relying on the domain knowledge of electrocardiography, we showed how to overcome the overfitting problem due to the bias we found in the employed dataset, thus by leveraging the proper ST-segments.



(a)



(b)

Figure 4.1: (a) Lasso weights and average QRST template for record patient005/s0025lre. Red dots point to the 10 largest weights. (b) Population averaged QRST template ± standard deviation for HC and anterior STEMI, computed over the PTB dataset.

To the best of our knowledge, a few works presented in Section 2.2.6 faced the explainability issue in the context of STEMI classification. Strodthoff *et al.* [91] explained a CNN classifier relying on the "gradient × input" method [196] and, in line with our results, the authors noticed that the most relevant segments for classification were located on the QRS complex. However, the employed method was designed only for CNN models and it cannot be generalized to other ML algorithms. Furthermore, the authors did not carry a dataset level analysis and they provided an explanation only for a few ECG test samples. On the other hand, Zhang *et al.* [94] employed SHAP [198] which is a model-agnostic, *post-hoc*, and local explainability method, and they performed a dataset level analysis. The authors explained the ECG leads that were most relevant in the classification of STEMI for a CNN architecture, by averaging the computed explanations on the entire employed dataset. However, the authors did not consider the localization of the infarct when providing explanations, since they employed a unique STEMI classification label when training the CNN. Hence, as it may be expected, a wide range of leads resulted equally important in the final classification process, thus *blurring* the final explanation. Finally, Ibrahim *et al.* [201] highlighted the clinical features that mostly contributed to the classification of myocardial infarction with an eXtreme Gradient Boosting (XGBoost) model [45] by means of Shapley values [199]. The authors obtained findings similar to the ones we reached, since they demonstrated that QRS related features were surprisingly relevant in the classification of STEMI, along with other not clinically related features (even if the authors did not rely on the same PTB dataset). However, even if the authors extracted clinical related features to STEMI, they only explained two sample ECGs and they still did not provide a dataset level analysis.

We addressed the drawbacks of the previous research works by employing LIME, which is a model-agnostic method that it capable of explaining the decision of any ML classifier, using a linear surrogate explanation model. Furthermore, we properly framed the explanations in the domain knowledge of electrocardiography by designing a custom metric which allowed us to highlight the importance of each

lead in the final classification output, and we averaged the explanations over all the training ECGs to avoid potentially cherry picked explanations. However, LIME presents some disadvantages that should be considered and tackled: 1) the correct definition of the kernel can sometimes be difficult, since it must be experimentally tested. 2) The sampling strategy for the creation of the artificial samples must be designed relying on domain knowledge. 3) Some works evidenced that explanations can be unstable, *e.g.* Alvarez-Melis *et al.* [225] showed that explanations of very close instances greatly vary in a simulated setting. To tackle this potential issue, we introduced an average explanation measure which is based on the entire training set.

To conclude, LIME may be considered a good ally in supporting researchers aiming to create automatic classifiers. We conclude that RF models can be not trustable as they can exploit not significant features for STEMI classification. A way to prevent such behavior is represented by LIME, which allowed us to see which parts of the input mostly contributed to the final classification. Our experiments suggests that in the case of small-size databases, a domain-knowledge based feature engineering and LIME can help in designing trustable classifiers, which rely on trustable features, such as STE.

4.3 Explainability of Deep Learning Algorithms in the Classification of 27 Cardiac Abnormalities

4.3.1 Introduction

As we already described in Section 1.3, the main advantage, and thus the reason of great popularity, of Deep Neural Network (DNN) models is represented by the optimal feature representation achieved after the training phase. The capability of automatic learning the relevant features is due to the large amount of parameters that these models contain (in the order of *tens of millions* [74]). However, we reported in Section 1.4 that with such large amount of parameters, the classification outputs of DNN models become difficult (or even impossible in certain situations) to understand. Consequently, automatic ECG classifications performed using DNN

models result difficult to be associated with a physiological interpretation. The perceived lack of interpretability gives the feeling of dealing with black-boxes: the process by which the models perform the classification can be inscrutable to humans, limiting the trust in them, and thus hindering the acceptance in the healthcare community [78, 79, 86].

In order to understand how DL methods provide classification outcomes, in Bodini *et al.* [111] we introduced two explainability frameworks specifically designed for CNN architectures trained to classify CA from ECGs, that let us to inspect the decision of a CNN by unveiling which waves of the input ECG were most relevant to the final classification outcome. In the mentioned work, we refer to the P wave, QRS complex and T wave composing the ECG beat as simply "waves". The rationale behind the development of new explainability frameworks in this context relies on the fact that the evaluation of the explanation itself results challenging with most of the currently available methodologies. Indeed, most explainability methods can highlight the most important samples of the ECG contributing to the final classification. However, understanding whether such samples result meaningful for the CA to detect is often not currently handled (see Section 2.2.6). For instance, a single heartbeat on 1-lead ECG sampled at $1000Hz$ has approximately 600 samples, and each sample has a weight on the classification. However, for a given abnormality, only some ECG samples must result useful for the classification, based on prior knowledge from electrocardiography.

In order to frame explanations in the knowledge domain, we combined two modules: the first one provides explainability by using two state-of-the-art techniques. The second one assesses whether the most important samples are matching those expected to be affected by the cardiac abnormality, thus including the domain knowledge. Differently from other domains, like Computer Vision [226], this approach results feasible because segmenting ECGs is rather easy in the considered context, with very well-established and validated algorithms already available.

4.3.2 Explainability Frameworks

We introduce two new frameworks to explain the classification results of CNN models, trained for the task of multi-label 12-lead ECG classification of CA. Both the frameworks comprise two modules. The first one relies on two model-specific, *post-hoc*, and local explainability algorithms, whose output shows the contribution of each ECG time sample to the final classification. The second one segments the ECG relying on validated algorithms (see Section 4.3.3) and it quantifies whether the ECG samples most relevant for classification belong to the ECG waves which the domain knowledge links to the CA.

Framework 1: Occlusion Method

The first framework uses an *occlusion-based methodology* [227]. It is an inspection technique originally designed to explain CNN for image classification, and we adapted this technique to ECGs. Given a 12-lead ECG, an occlusion is performed by setting to zero a specific interval of the signal (*e.g.* all the samples in the T wave are set to zero). The CNN classifier is then run to compute the output class after applying the occlusion. The occlusion of the segments that leads to a relevant change in the classification output, with respect to the ground truth labels, points to those segments which are important for the final classification. In particular, in this work the occlusion was performed by setting to zero, for each beat and lead, all the samples relative to either the P wave, QRS complex, or T wave.

Then, we calculated the percentage variation in the model output of a given class after the occlusion of the three waves. Finally, we normalized these three values for their sum. We termed these three normalized quantities as *Relevance* (RV) measures [98], that we formally defined for each ECG $\boldsymbol{x}$ as

$$RV_{c,w}^{\mathrm{F1}}(\boldsymbol{x}) = \frac{|P_c(\boldsymbol{z}_w \circ \boldsymbol{x}) - P_c(\boldsymbol{x})|}{\sum_{i\in\{\mathrm{P,QRS,T}\}} |P_c(\boldsymbol{z}_i \circ \boldsymbol{x}) - P_c(\boldsymbol{x})|}, \tag{4.2}$$

where $\boldsymbol{z}_w$ is a mask vector containing ones in the position of the indices belonging to the wave w in the ECG (*i.e.* P, QRS or T wave), $\circ$ is the element-wise product, and $P_c(\boldsymbol{x})$ is the probability estimated by the CNN for the considered class c.

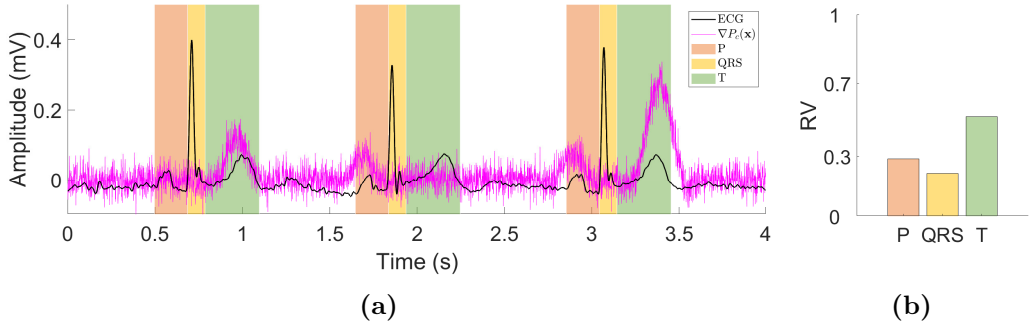
Framework 2: Saliency Maps

The second framework implements *saliency maps* in the first module: the CNN output $P_c(\boldsymbol{x})$ for the class c and the ECG input $\boldsymbol{x}$ is approximated with a first-order Taylor expansion in the form $P_c(\boldsymbol{x}) \approx \boldsymbol{m}^\intercal \boldsymbol{x} + q$, where q is a scalar quantity and $\boldsymbol{m} = \nabla P_c(\boldsymbol{x})$ is the weight vector [228]. The latter stands as the explanation for the classification of the underlying CNN, since the largest entries of $\boldsymbol{m}$ are associated to the samples that are the most relevant in the final classification.

The RV value is quantified as follows. For each segmented beat, we computed the sum of the absolute value of the weights belonging to any of the three ECG waves. Then, we averaged these three values across the beats of a given signal. The three values, obtained for any ECG, were normalized on the length of the ECG waves (190ms, 100ms and 310ms for the P, QRS, and T, respectively) and to have unit sum. Differently from the first framework, this approach weights the RV based on value of the entries of the vector $\boldsymbol{m}$. The formulation of RV for the input signal $\boldsymbol{x}$ in this second framework is

$$RV_{c,w}^{\mathrm{F2}}(\boldsymbol{x}) = \frac{(\boldsymbol{z}_w^\intercal \boldsymbol{z}_w)^{-1} \boldsymbol{z}_w^\intercal |\nabla P_c(\boldsymbol{x})|}{\sum_{i \in \{\mathrm{P,QRS,T}\}} (\boldsymbol{z}_i^\intercal \boldsymbol{z}_i)^{-1} \boldsymbol{z}_i^\intercal |\nabla P_c(\boldsymbol{x})|}, \tag{4.3}$$

where $\boldsymbol{z}_w$ is the same mask vector of Equation 4.2, and $|\nabla P_c(\boldsymbol{x})|$ is the absolute value of the gradient of $P_c(\boldsymbol{x})$ (column vector). Figure 4.2 reports an example of explanation provided by this second framework on a single-lead ECG.



(a) **(b)**

Figure 4.2: Example of explanation for a 5s ECG by means of Framework 2. For a given ECG (black line), the output of the first module of the framework is reported, for each ECG sample, in magenta (a). The shaded boxes represent the segmentation of each ECG wave. Then, the second module computes the RV of the P wave, QRS complex and T wave (b). In this example, the T wave was linked to the largest RV, as shown in (b).

4.3.3 Preprocessing of the Electrocardiograms

Before the execution of experiments, the 12-lead ECGs were downsampled or upsampled to $500Hz$ according to their actual sampling rate and filtered with a bandpass Butterworth filter (3^{rd} order, zero phase, and pass-band: $0.67 - 30Hz$) to reduce powerline interference, baseline wandering and high frequency noise. Beat detection was performed on the vector magnitude signal (VM) relying on the *gqrs* algorithm [46] and beat positions, *i.e.* the R-peaks, were refined using the Woody algorithm applied to the VM [202]. Within each signal, we segmented the P wave, QRS complex and T wave for all beats. Segments relied on the R peaks previously identified and were defined as follows: 1) P wave: R$-240ms$ to R$-50ms$; 2) QRS complex: R $\pm 50ms$; 3) T wave: R$+50ms$ to R$+360ms$.

4.3.4 The Experimental Settings

The two designed frameworks were tested on a CNN model trained for multi-class classification of CA from 12-lead ECGs, specifically developed for the 2020 PhysioNet/Computing in Cardiology challenge [68]. We extensively presented the dataset employed in the context of the challenge in Section 3.2.2.

We selected the CNN model that won the official phase of the challenge, namely the BUTTeam network [229], which is a residual neural network inspired by He *et al.* [73]. The main rationale behind the selection of this model was that the good performance of the network was certified by the official ranking of all submitted models, and such ranking was built considering an unseen test set. We did not retrain the model from scratch, but we instead used the pre-trained model provided by the authors (available at `https://github.com/tomasvicar/BUTTeam`).

The performance of the CNN on the entire challenge dataset (see Section 3.2.2) was quantified using the macro measures AUC, AUPRC, F1 and Accuracy. Macro measures refer to the average of all values of a given measure determined for each class. The AUC is computed as the area under the curve defined by true positive rate (TPR)/sensitivity and true negative rate (TNR)/specificity. Similarly, the AUPRC is computed as the area under the curve depicted by TPR/sensitivity

and PPV/precision. F1 is computed as the product of twice PPV/precision and TPR/sensitivity normalized by their sum. Accuracy is the proportion of correctly classified ECG over the total number of ECGs. The performance of the model are 0.67, 0.44, 0.50 and 0.42 for AUC, AUPRC, F1, and Accuracy, respectively. We finally disclose that we merged the classes which the committee scored as the same diagnosis, thus we finally considered 24 classes instead of 27 (see Section 3.3).

In these experiments, we quantified the average RV across all signals of a given class and the confidence interval of the mean at $1 - \alpha = 0.95$ confidence level. It is worth recalling that, differently from other explainability methodologies, such quantification was possible because of the beat segmentation performed on the ECG.

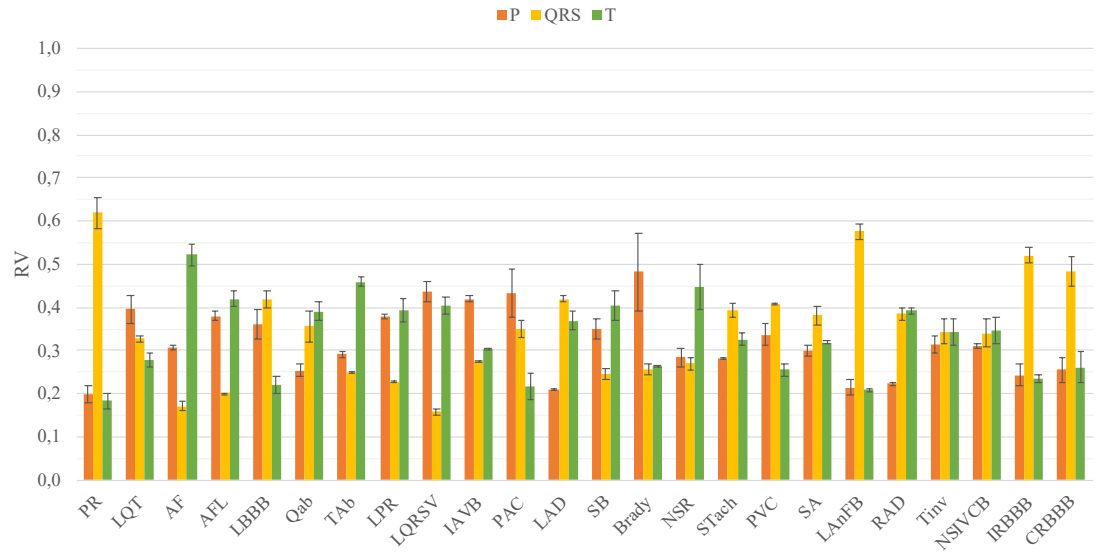
In addition, we determined whether the two explainability frameworks were in agreement between each other by means of the Hellinger distance [230], quantifying the agreement as follows

$$a_c = 1 - \frac{1}{\sqrt{2}} \left[\sum_{i \in \{\mathrm{P,QRS,T}\}} \left(\sqrt{\overline{RV}^{\mathrm{F1}}_{c,i}} - \sqrt{\overline{RV}^{\mathrm{F2}}_{c,i}} \right)^2 \right]^{\frac{1}{2}}, \tag{4.4}$$

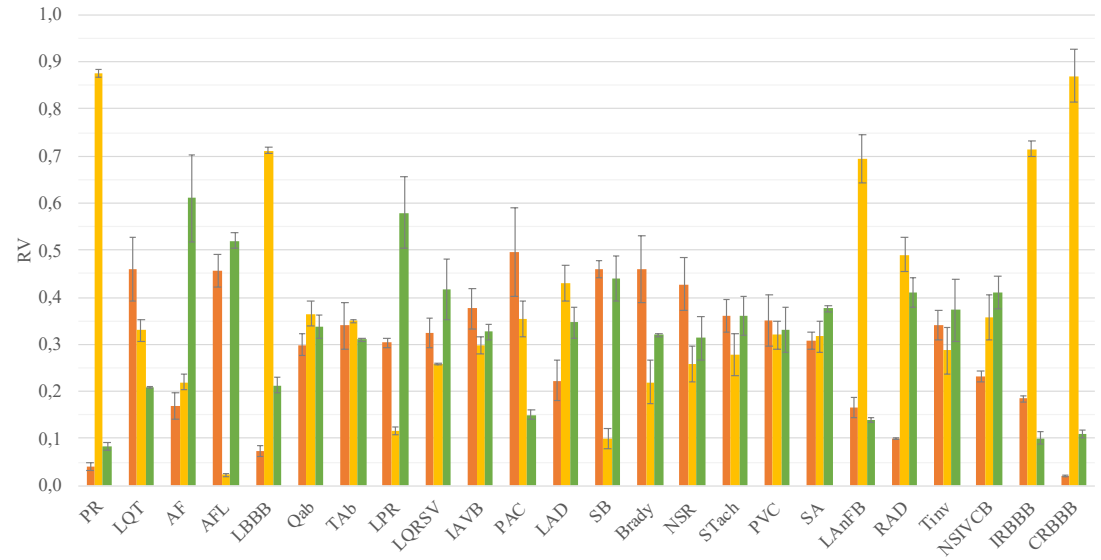
where $\overline{\mathrm{RV}}^{\mathrm{F1}}_{c,i}$ and $\overline{\mathrm{RV}}^{\mathrm{F2}}_{c,i}$ were the average RV over signals for class c and wave i, for Framework 1 and 2, respectively. The agreement for the class c is maximum when the two triplets of RV values are equal, whereas it is minimum when the frameworks point to different waves with maximum RV. The agreement was calculated for each class.

4.3.5 Experimental Results

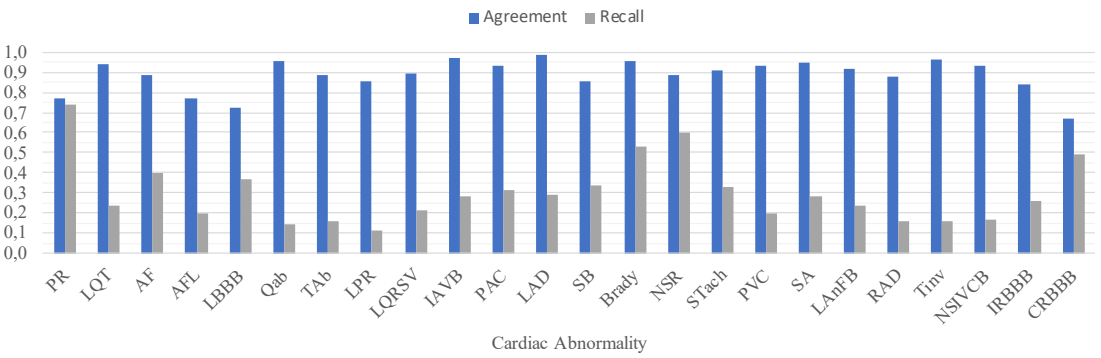
In Figure 4.3(a) and 4.3(b), we report the average RV and their confidence intervals for each class (CA) considered by the classifier, in Framework 1 panel (a), and 2 panel (b), computed relying on Equation 4.2 and 4.3, respectively. The acronyms related to the classes were reported in Section 3.2.2. Focusing on the comparison between the confidence intervals, for Framework 1 classes PR, AF, PAC, Brady, NSR, LAnFB, IRBBB, and CRBBB displayed average values of maximum RV which were significantly different between QRS, T and P (hinting that one of these regions was significantly linked to the classifier's output). Similarly, PR,



(a)



(b)



(c)

Figure 4.3: Average and confidence interval of *RV*, for each class and ECG wave of both the frameworks, *i.e.* occlusion (a) and saliency maps (b). Agreement between the two methodologies and recall for each cardiac abnormality (c).

LQT, AF, LBBB, LPR, Brady, LAnFB, RAD, IRBBB, and CRBBB if the case of Framework 2. For eight classes, both frameworks considered the same ECG wave as the most relevant for the classification.

In Figure 4.3(c), we report the agreement plot between Framework 1 and 2, which is in general very high. The three classes with the highest agreement are IAVB, LAD, and Tinv. In such cases, the explainability methods agree that the most relevant part of the ECG to take into consideration are the P wave (IAVB), QRS complex (LAD) and T wave (Tinv). The three classes with the lowest agreements are PR, LBBB and CRBBB. For only 4 out of 24 cardiac abnormalities (classes) the agreement is below 80%.

4.3.6 Discussion

The main contribution of our study is threefold. First, both the frameworks were capable to provide a local explanation for a given ECG classified by the CNN, by suggesting which ECG wave was involved in the decision. Depending on the class-wise recognition accuracy (recall) achieved and the ECG wave highlighted by the framework, our understanding of the decision process in place by the network may change. In fact, when the recall is low and the framework points to the right ECG wave for the diagnosis of the cardiac abnormality, the network is likely confounding its decisions with other abnormalities involving the same wave. For example, high RV values regarding the T wave may mean that the network did not learn the correct pattern to distinguish between T wave abnormality or T wave inversion, but understood to focus on the correct wave. On the other hand, when the recall is high and the RV value is low, the network might be overfitting on the given dataset, or the cardiac abnormality is not uniquely related to a specific ECG wave (*e.g.* AF). In our work [98] presented in Section 4.2, using the LIME methodology [183] we found that a RF classifier was mostly relying on the QRS peak amplitude for providing its classification between normal ECG vs myocardial infarction. Such feature is not related with the considered cardiac abnormality, hence the ML algorithm was overfitting on the current dataset (which was then confirmed by further analysis).

For the two remaining cases, *i.e.* high recall with high RV and low recall with low RV, the explanations are straightforward: the network has either learned to properly distinguish the cardiac abnormality from the others, or simply not.

Second, comparing the results of the framework with the domain knowledge. It became possible to determine whether the DL model selected was, on average, focusing on the right ECG wave. For example, a case when the ML method agreed with the cardiology domain knowledge was CRBBB (complete right bundle branch block) which affects the QRS complex morphology: both frameworks reported a maximum RV in correspondence with this ECG wave (Fig. 4.3a and 4.3b). On the contrary, for other classes the agreement with the domain knowledge was minimal. It is the case of Qab (abnormal Q point), where the Q point of the QRS complex is not within normality range: the frameworks pointed out that the model did not focus on any particular portion of the ECG, resulting in similar RV values for QRS, T and P waves. Similar results were achieved for Tinv (T wave inversion).

Third, the frameworks agreed with each other for most of the classes (Fig. 4.3c), while differences were found for a few cardiac abnormalities. One may expect that the differences in the agreement between frameworks may be connected with the recall of the specific class. In other words, when the recognition of the network is low, the agreement is also low, and viceversa. However, such expectation seems not to be supported by our findings (Figure 4.3c). Indeed, we found that the agreement was unrelated with the class-wise performance of the network (Pearson's correlation coefficient between a_c and recall for each class was -0.4; $p > 0.05$). A similar result was obtained in a recent contribution from the Computer Vision domain, where several Explainable AI techniques were found to have significantly different performance (with the occlusion method as ground truth), even with a state-of-the-art DNN trained on millions of images [231].

Other recent studies proposed algorithms for explaining the decisions of DNN models for automatic ECG classification, as we saw in Section 2.2.6. Several of the reported works limited to highlight the samples belonging to a single ECG beat that mostly contributed to the classification, without framing the explanations into

the domain knowledge. On the other hand, other methods were able to point which wave (P, QRS, T), beat, or combination of beats were important for the classification outcome. There are mainly three differences between the reported works and our approach. First, to the best of our knowledge, we are the first at the same time with Zhang *et al.* [94], to systematically evaluate the performance of a DNN against the domain knowledge of ECG interpretation. Our frameworks indeed not only provide the ECG samples important for the classification, but also the importance of each wave in the final decision by means of the measure RV. Second, the evaluation was performed on 24 different CA, while most of related works only considered a few classes. Third, our frameworks are also suitable for models already trained, and thus they do not necessarily require a dataset to be executed. We finally notice that the mentioned work of Zhang *et al.* [94] highlighted the contribution of each lead in the final classification outcome when explaining a CNN. As we already mentioned in Section 2.2.6, on the contrary with respect to our work, most of the CA selected in Zhang *et al.* [94] can be observed on any of the 12 leads. On the other hand, we framed the explanations by considering ECG waves which are potentially affected by certain CA, regardless of the considered lead, in the most of the CA we considered.

Both frameworks can be considered from two different perspectives: the one of the ML expert (who creates the classifier) and the one of the physician (who uses the tool in the clinical practice). From the perspective of the ML expert, the frameworks allow to inspect if the CNN relies on the ECG segments expected for the classification according to the clinical standard practice. Otherwise, the user can try to address the issue by understanding the reasons behind it, and thus guiding the architecture towards the domain knowledge. From the perspective of the physician, the frameworks allow to understand whether the decision taken relied on a known domain knowledge by highlighting the corresponding ECG wave, thus the trust in the DL model can be increased.

Under a supervised classification framework, a ML algorithm take a decision that is represented by the classification output itself. As a consequence, ML decision-makers can be trusted relying only on their predictive performance evaluated on

the dataset available. Our effort in the direction of developing an explainability methodology relies on the fact that we believe necessary that future advancement in automatic processing of ECG progresses together with our capability of understanding the decisions taken by a ML model. In this way, the large accuracy which ML algorithms might obtain in the future will also contribute in progressing the understanding of the underlying physiology.

Similar considerations were already present in the thinking of the ancient Greeks to obtain what Aristotle defined as τέχνη [téchne]: a real productive science [87]. He noted that the technological advancement can be obtained with different means, that could be achieved by either scientists or empirics. However, Aristotle set apart scientists from the empirics: people with a high degree of expertise in a specific domain, but who lack of any theory to justify their results. The empirics can even often achieve outstanding results, but what clearly divide science from empiricism is that it comes with a theory whose domain principles justify why certain decisions are correct in specific circumstances [78, 87]. ML algorithms are like empirics to a certain extents, but complemented with means of understanding (explaining) their decision process may lead to scientific knowledge.

In our opinion, we do not believe that the current state-of-the-art ML algorithms might outperform significantly the human capability of detecting cardiac abnormalities. In fact, the number of possible confounding factors, co-morbidities, number of rare conditions, and evolution in time of the diseases may all increase the amount of data necessary for ML models to be trained. In order to mitigate such issues, the creation of innovative ML algorithms, capable of incorporating the domain knowledge, would facilitate the development of these models (*e.g.* less data-hungry algorithms, faster training, larger explainability) and the introduction of such methodologies in the clinical practice, thus fostering trust for their use.

4.3.7 Limitations of the Study

The frameworks presented some limitations. First, rhythm-based cardiac abnormalities were not properly handled by our methodology. Given the fact that an altered

rhythm may or may not affect the regularity of the occurrence of any waves, it was not possible to define the one-to-one match between the CA and a specific ECG wave. For example, Brady, one of the class detected correctly by the CNN (recall of 0.53), refers to a very low heart rate. Typically, heart-rate alterations are quantified looking at the time intervals between consecutive R peaks because of their ease of detection. However, the frameworks found the P wave very important for this class (Figure 4.3a). Given the fact that the P wave does not change during bradycardia, the assessment of the low heart rate might have been performed by the network "looking" at the rate of both P wave and QRS complex. Similarly, the frameworks found the T wave relevant for the detection of AF, which is characterized by the absence of the P wave, an oscillatory pattern on the ECG baseline and irregular heart rate. In this case, the frameworks hint that the CNN may use samples between consecutive beats where the T wave, the isoelectric line, and part of the P wave are located to detect AF. The same observation is shared by the work of Mousavi et al. [93], where an attention mechanism was used to show that the network relied on samples between consecutive beats to detect AF. Third, PVC have a morphology which is largely different from a normal sinus beat. Therefore, considering portion of the ECG where the P, QRS and T waves are usually located was not relevant (the different ECG waves had a similar low RV value).

Even if the algorithms for ECG segmentation have become well-established in the recent years, it must be noted that the performance of the frameworks is potentially dependent on the algorithm used. Usually, segmentation is performed after beat detection which is dependent on the quality of the ECG [232]. We did not focus on the selection of the most robust segmentation algorithm for the clinical 12-lead ECG. However, since this type of ECG can be acquired at low cost, physicians usually recollect measurements in case of low quality. We therefore assumed that the ECG within the dataset were of sufficient quality (but we leave this investigation for the future). On the other hand, when the ECG is acquired in different contexts, *e.g.* sport activities and Holter acquisitions, the quality could be lower. In such cases, a careful preprocessing should be applied before running our frameworks.

5
Conclusion

Thanks to the high computational resources and growing availability of ECG datasets, the application of ML and DL algorithms has been more and more widely investigated in the context of classification of CA from ECGs. However, despite the promising performance reported in several research works, the development of ML, and especially DL, algorithms within this context are still in their *infancy* stage, and in our view there are still several challenges to be addressed before its future clinical usage.

The process of ECG standardization is crucial in the context of classification of CA from ECGs, however there is still no standard ECG input format or data preprocessing protocol. As we discussed in Section 2.1, despite the clinical ECG lasts $10s$, many research works proposed ML and DL models with custom dimensionality (*e.g.* single-beat, single-lead, *etc.*). This poses the problem of comparing the performance accross studies. Indeed, we feel that the problem of ECG data standardization is underestimated in the context of classification of CA from ECGs. Furthermore, non-clinical ECGs (collected from Holter, wearable devices, *etc.*) still need further investigation to reliably identify CA that are intrinsically transient, for which these sensing modalities are meant of.

Reproducibility and generalizability pose another relevant barrier that must be addressed before applying ML and DL algorithms to the standard clinical practice.

Especially in the context of DL, the classification performance greatly depends on the amount and quality of ECGs employed to train algorithms, which might be often inconsistent. However, as we discussed in Section 1.4, most research works limit to collect ECG training data only from a single center, few acquisition devices, and limited populations, or they leverage publicly available unbalanced ECG datasets which were collected for specific clinical purposes, hence possible containing potential biases (as we discussed in Section 4.2). Even if we attempted to address the mentioned problems throughout the thesis work, we think that in the following research it will be necessary to properly assess the classification performance of ML and DL algorithms by employing more and more wider and validated ECG datasets to avoid the risk of overestimating the capability of such algorithms.

Explainability still remains one of the crucial problems to be addressed to fully achieve the trust of physicians and insert ML and DL algorithms into their standard clinical workflow. However, the current ML algorithms, and especially the DL ones, are essentially developed to directly output classification outcomes without providing any reason about the employed process, thus turning such systems into black-boxes. Despite we tried to address the problem of explainability in Chapter 4, the investigation of explainable ML and DL algorithms for ECG classification is currently in a preliminary stage, and most of the introduced research works are still under ongoing investigation.

As a final conclusion, we believe that despite the fast advancements of ML and DL, there is nowadays no evidence hinting that the role of expert physicians will ever be removed in ECG interpretation. In our view, ML and DL algorithms must be designed only to help the process of ECG interpretation carried by physicians, as adjunct decision support systems, instead of fully replacing their role. As it must be in any area of expertise, the central role of highly trained physicians in the context of ECG interpretation will remain immovable in the foreseeable future.

References

[1] T. Gaziano et al. “Cardiovascular Disease”. In: *Disease Control Priorities in Developing Countries*. Ed. by D.T. Jamison et al. 2nd ed. Washington D. C., United States: The International Bank for Reconstruction and Development / The World Bank, 2006, pp. 645–662.

[2] S. Mendis et al. *Global atlas on cardiovascular disease prevention and control*. Geneve, Switzerland: World Health Organization, 2011.

[3] S. J. George and C. Lyon. “Pathogenesis of Atherosclerosis”. In: *Atherosclerosis: Molecular and Cellular Mechanisms*. Ed. by S. J. George and J. Johnson. Hoboken, NJ, United Stated: John Wiley & Sons, Ltd, 2010, pp. 1–20.

[4] T. L. Stedman. *Stedman’ s Medical Dictionary for the Health Professions and Nursing*. 7th ed. Philadelphia, PA, United States: Lippincott Williams & Wilkins, 2011.

[5] A. J. Marelli et al. “Congenital Heart Disease in the General Population”. In: *Circulation* 115.2 (2007), pp. 163–172.

[6] N. Lambert et al. “Rubella”. In: *The Lancet* 385.9984 (2015), pp. 2297–2307.

[7] G. M. Blue et al. “Congenital heart disease: current knowledge about causes and inheritance”. In: *Medical Journal of Australia* 197.3 (2012), pp. 155–159.

[8] E. Marijon et al. “Rheumatic heart disease”. In: *The Lancet* 379.9819 (2012), pp. 953–964.

[9] J. Brieler, M. A. Breeden, and J. Tucker. “Cardiomyopathy: An Overview”. In: *American Family Physician* 96.10 (2017), pp. 640–646.

[10] C. Antzelevitch and A. Burashnikov. “Overview of Basic Mechanisms of Cardiac Arrhythmia”. In: *Cardiac Electrophysiology Clinics* 3.1 (2011), pp. 23–45.

[11] A. L. Goldberger, Z. D. Goldberger, and A. Shvilkin. “Chapter 13 - Sinus and Escape Rhythms”. In: *Goldberger’s Clinical Electrocardiography*. 8th ed. Philadelphia, PA, United States: Saunders, 2012, pp. 114–120.

[12] H. V. Huikuri, A. Castellanos, and R. J. Myerburg. “Sudden Death Due to Cardiac Arrhythmias”. In: *New England Journal of Medicine* 345.20 (2001), pp. 1473–1482.

[13] V. C. Savona and V. Grech. “Concepts in cardiology - a historical perspective”. In: *Images in Paediatric Cardiology* 1.1 (1999), pp. 22–31.

[14] M. AlGhatrif and J. Lindsay. “A brief review: history to understand fundamentals of electrocardiography”. In: *Journal of Community Hospital Internal Medicine Perspectives* 2.1, 14383 (2012).

[15] H. P. Adams et al. “Guidelines for the Early Management of Adults With Ischemic Stroke”. In: *Stroke* 38.5 (2007), pp. 1655–1711.

[16] P. Kligfield et al. “Recommendations for the Standardization and Interpretation of the Electrocardiogram”. In: *Journal of the American College of Cardiology* 49.10 (2007), pp. 1109–1127.

[17] J. Hampton and J. Hampton. *The ECG Made Easy*. Philadelphia, PA, United States: Elsevier, 2019.

[18] L. Sörnmo and P. Laguna. “Chapter 2 - The Electroencephalogram — A Brief Background”. In: *Bioelectrical Signal Processing in Cardiac and Neurological Applications*. Ed. by L. Sörnmo and P. Laguna. New York, NY, United States: Academic Press, 2005, pp. 25–53.

[19] S. J. Stellpflug, J. S. Holger, and S. W. Smith. “What is the Role of the ECG in ACS?” In: *Critical Decisions in Emergency and Acute Care Electrocardiography*. Ed. by W. J. Brady and J. D. Truwit. Hoboken, NJ, United States: Wiley-Blackwell, 2009, pp. 83–91.

[20] M. Bickerton and A. Pooler. “Misplaced ECG electrodes and the need for continuing training”. In: *British Journal of Cardiac Nursing* 14.3 (2019), pp. 123–132.

[21] J. Schläpfer and H. J. Wellens. “Computer-Interpreted Electrocardiograms”. In: *Journal of the American College of Cardiology* 70.9 (2017), pp. 1183–1192.

[22] P. M. Rautaharju. “Eyewitness to history: Landmarks in the development of computerized electrocardiography”. In: *Journal of Electrocardiology* 49.1 (2016), pp. 1–6.

[23] D. Finlay et al. “Overview of featurization techniques used in traditional versus emerging deep learning-based algorithms for automated interpretation of the 12-lead ECG”. In: *Journal of Electrocardiology* (2021), in press.

[24] J. De Bie et al. “Performance of seven ECG interpretation programs in identifying arrhythmia and acute cardiovascular syndrome”. In: *Journal of Electrocardiology* 58 (2020), pp. 143–149.

[25] J. L. Willems et al. “Assessment of the performance of electrocardiographic computer programs with the use of a reference data base”. In: *Circulation* 71.3 (1985), pp. 523–534.

[26] R. H. Hongo and N. Goldschlager. “Status of Computerized Electrocardiography”. In: *Cardiology Clinics* 24.3 (2006), pp. 491–504.

[27] T. Novotny et al. “The role of computerized diagnostic proposals in the interpretation of the 12-lead electrocardiogram by cardiology and non-cardiology fellows”. In: *International Journal of Medical Informatics* 101 (2017), pp. 85–92.

[28] P. W. Macfarlane, B. Devine, and E. Clark. “The university of glasgow (Uni-G) ECG analysis program”. In: *Computers in Cardiology, 2005* (25–26 Sept. 2005, Lyon, France). Ed. by A. Murray. Vol. 32. New York, NY, USA: IEEE, pp. 451–454.

[29] E. R. Dorsey and E. J. Topol. “Telemedicine 2020 and the next decade”. In: *The Lancet* 395.10227 (2020), p. 859.

[30] A. L. P. Ribeiro et al. “Tele-electrocardiography and bigdata: The CODE (Clinical Outcomes in Digital Electrocardiography) study”. In: *Journal of Electrocardiology* 57 (2019), S75–S78.

[31] N. A. M. Estes. "Computerized Interpretation of ECGs: Supplement Not a Substitute". In: *Circulation: Arrhythmia and Electrophysiology* 6.1 (2013), pp. 2–4.

[32] H. Smulyan. "The Computerized ECG: Friend and Foe". In: *The American Journal of Medicine* 132.2 (2019), pp. 153–160.

[33] A. P. Shah and S. A. Rubin. "Errors in the computerized electrocardiogram interpretation of cardiac rhythm". In: *Journal of Electrocardiology* 40.5 (2007), pp. 385–390.

[34] R. M. Farrell, J. Q. Xue, and B. J. Young. "Enhanced rhythm analysis for resting ECG using spectral and time-domain techniques". In: *Computers in Cardiology, 2003* (21–24 Sept. 2003, Thessaloniki, Greece). Ed. by A. Murray. New York, NY, USA: IEEE, pp. 733–736.

[35] N. Faruk et al. "A comprehensive survey on low-cost ECG acquisition systems: Advances on design specifications, challenges and future direction". In: *Biocybernetics and Biomedical Engineering* 41.2 (2021), pp. 474–502.

[36] S. Gaube et al. "Do as AI say: susceptibility in deployment of clinical decision-aids". In: *npj Digital Medicine* 4, 31 (2021).

[37] N. Kagiyama et al. "Artificial Intelligence: Practical Primer for Clinical Research in Cardiovascular Disease". In: *Journal of the American Heart Association* 8.17, e012788 (2019).

[38] S. Hong et al. "Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review". In: *Computers in Biology and Medicine* 122, 103801 (2020).

[39] A. K. Feeny et al. "Artificial Intelligence and Machine Learning in Arrhythmias and Cardiac Electrophysiology". In: *Circulation: Arrhythmia and Electrophysiology* 13.8 (2020), e007952.

[40] K. C. Siontis et al. "Artificial intelligence-enhanced electrocardiography in cardiovascular disease management". In: *Nature Reviews Cardiology* 18.7 (2021), pp. 465–478.

[41] R. K. Sevakula et al. "State-of-the-Art Machine Learning Techniques Aiming to Improve Patient Outcomes Pertaining to the Cardiovascular System". In: *Journal of the American Heart Association* 9.4, e013924 (2020).

[42] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. 4th ed. Hoboken, NJ, USA: Pearson, 2020.

[43] A. Rajkomar, J. Dean, and I. Kohane. "Machine Learning in Medicine". In: *New England Journal of Medicine* 380.14 (2019), pp. 1347–1358.

[44] C. M. Bishop. *Pattern Recognition and Machine Learning*. 1st ed. New York, NY, USA: Springer-Verlag, 2006.

[45] A. Burkov. *The hundred-page machine learning book*. Québec City, Canada: Andriy Burkov, 2019.

[46] A. L. Goldberger et al. "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals". In: *Circulation* 101.23 (2000), e215–e220.

[47] A. Mincholé et al. “Machine learning in the electrocardiogram”. In: *Journal of Electrocardiology* 57 (2019), S61–S64.

[48] L. Xie et al. “Computational Diagnostic Techniques for Electrocardiogram Signal Analysis”. In: *Sensors* 20.21, 6318 (2020).

[49] X. Liu et al. “Deep learning in ECG diagnosis: A review”. In: *Knowledge-Based Systems* 227.5, 107187 (2021).

[50] N. Saghir et al. “A comparison of manual electrocardiographic interval and waveform analysis in lead 1 of 12-lead ECG and Apple Watch ECG: A validation study”. In: *Cardiovascular Digital Health Journal* 1.1 (2020), pp. 30–36.

[51] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning”. In: *Nature* 521.7553 (2015), pp. 436–444.

[52] J. Schmidhuber. “Deep learning in neural networks: An overview”. In: *Neural Networks* 61 (2015), pp. 85–117.

[53] F. Emmert-Streib et al. “An Introductory Review of Deep Learning for Prediction Models With Big Data”. In: *Frontiers in Artificial Intelligence* 3, 4 (2020).

[54] S. Somani et al. “Deep learning and the electrocardiogram: review of the current state-of-the-art”. In: *EP Europace* 23.8 (2021), pp. 1179–1191.

[55] Z. Ebrahimi et al. “A review on deep learning methods for ECG arrhythmia classification”. In: *Expert Systems with Applications: X* 7, 100033 (2020).

[56] L. Liu et al. “Deep Learning for Generic Object Detection: A Survey”. In: *International Journal of Computer Vision* 128 (2019), pp. 261–318.

[57] D. Ciregan, U. Meier, and J. Schmidhuber. “Multi-column deep neural networks for image classification”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition* (16–21 June 2012, Providence, RI, USA). New York, NY, USA: IEEE, pp. 3642–3649.

[58] Y. Taigman et al. “DeepFace: Closing the Gap to Human-Level Performance in Face Verification”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition* (23–28 June 2014, Columbus, OH, USA). New York, NY, USA: IEEE, pp. 1701–1708.

[59] L. Wan et al. “Regularization of Neural Networks using DropConnect”. In: *Proceedings of the 30th International Conference on Machine Learning* (26–21 June 2013, Atlanta, GA, USA). Ed. by S. Dasgupta and D. McAllester. Vol. 28. 3. Atlanta, GA, USA: PMLR, pp. 1058–1066.

[60] Matteo Bodini. “A Review of Facial Landmark Extraction in 2D Images and Videos Using Deep Learning”. In: *Big Data and Cognitive Computing* 3.1, 14 (2019).

[61] K. He et al. “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *2015 IEEE International Conference on Computer Vision (ICCV)* (7–13 Dec. 2015, Santiago, Chile). New York, NY, USA: IEEE, pp. 1026–1034.

[62] O. Russakovsky et al. “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252.

[63] J. Zheng et al. “A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients”. In: *Scientific Data* 7.1, 48 (2020).

[64] P. Wagner et al. “PTB-XL, a large publicly available electrocardiography dataset”. In: *Scientific Data* 7.1, 154 (2020).

[65] R. Bousseljot, D. Kreiseler, and A. Schnabel. “Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet”. In: *Biomedizinische Technik* 40.1 (1995), pp. 317–318.

[66] D. Kreiseler and R. Bousseliot. “Automatisierte EKG-Auswertung mit Hilfe der EKG-Signaldatenbank CARDIODAT der PTB”. In: *Biomedizinische Technik* 40.1 (1995), pp. 319–320.

[67] A. Y. Hannun et al. “Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network”. In: *Nature Medicine* 25.1 (2019), pp. 65–69.

[68] E. A. Perez Alday et al. “Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020”. In: *Physiological Measurement* 41.12, 124003 (2021).

[69] M. A. Reyna et al. “Will Two Do? Varying Dimensions in Electrocardiography: the PhysioNet/Computing in Cardiology Challenge 2021”. In: *2020 Computing in Cardiology* (12–15 Sept. 2021, Brno, Czech Republic). Vol. 48. New York, NY, USA: IEEE, pp. 1–4.

[70] H. Zhu et al. “Automatic multilabel electrocardiogram diagnosis of heart rhythm or conduction abnormalities with deep learning: a cohort study”. In: *The Lancet Digital Health* 2.7 (2020), e348–e357.

[71] A. H. Ribeiro et al. “Automatic diagnosis of the 12-lead ECG using a deep neural network”. In: *Nature Communications* 11.1, 1760 (2020).

[72] M. B. Alkmim et al. “Improving patient access to specialized health care: the Telehealth Network of Minas Gerais, Brazil”. In: *Bulletin of the World Health Organization* 90.5 (2012), pp. 373–378.

[73] K. He et al. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (27–30 June 2013, Las Vegas, NV, USA). New York, NY, USA: IEEE, pp. 770–778.

[74] S. Zagoruyko and N. Komodakis. “Wide Residual Networks”. In: *Proceedings of the British Machine Vision Conference 2016* (19–22 Sept. 2016, York, United Kingdom). Ed. by R. C. Wilson, E. R. Hancock, and W. A. P. Smith. Durham, United Kingdom: BMVA Press, pp. 87.1–87.12.

[75] R. Guidotti et al. “A Survey of Methods for Explaining Black Box Models”. In: *ACM computing surveys* 51.5, 93 (2018).

[76] N. Burkart and M. F. Huber. “A Survey on the Explainability of Supervised Machine Learning”. In: *Journal of Artificial Intelligence Research* 70 (2021), pp. 245–317.

[77] C. Molnar. *Interpretable Machine Learning*. Victoria, Canada: Leanpub, 2020.

[78] A. J. London. “Artificial Intelligence and Black-Box Medical Decisions: Accuracy versus Explainability”. In: *Hastings Center Report* 49.1 (2019), pp. 15–21.

[79] C. Rudin. “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead”. In: *Nature Machine Intelligence* 1.5 (2019), pp. 206–215.

[80] K. R. Varshney and H. Alemzadeh. “On the Safety of Machine Learning: Cyber-Physical Systems, Decision Sciences, and Data Products”. In: *Big Data* 5.3 (2017), pp. 246–255.

[81] S. Levin and J. C. Wong. “Self-driving Uber kills Arizona woman in first fatal crash involving pedestrian”. In: *The Guardian* (Mar. 2018).

[82] C. Said. “Video shows Uber robot car in fatal accident did not try to avoid woman”. In: *The Guardian* (Mar. 2018).

[83] T. Grote and P. Berens. “On the ethics of algorithmic decision-making in healthcare”. In: *Journal of Medical Ethics* 46.3 (2020), pp. 205–211.

[84] A. Esteva et al. “A guide to deep learning in healthcare”. In: *Nature Medicine* 25.1 (2019), pp. 24–29.

[85] Eric J. Topol. “High-performance medicine: the convergence of human and artificial intelligence”. In: *Nature Medicine* 25.1 (2019), pp. 44–56.

[86] F. Wang, R. Kaushal, and D. Khullar. “Should Health Care Demand Interpretable Artificial Intelligence or Accept “Black Box” Medicine?” In: *Annals of Internal Medicine* 172.1 (2020), pp. 59–60.

[87] J. Barnes, ed. *Complete Works of Aristotle, Volume 1: The Revised Oxford Translation*. Princeton, NJ, United States: Princeton University Press, 1984.

[88] A. J. London. “Moral Knowledge and the Acquisition of Virtue in Aristotle’s “Nicomachean” and “Eudemian Ethics””. In: *The Review of Metaphysics* 54.3 (2001), pp. 553–583.

[89] W. Roberts et al. “Across the centuries: Piecing together the anatomy of the heart”. In: *Translational Research in Anatomy* 17, 100051 (2019).

[90] H. Xintian et al. “Deep learning models for electrocardiograms are susceptible to adversarial attack”. In: *Nature Medicine* 26.3 (2020), pp. 360–363.

[91] N. Strodthoff and C. Strodthoff. “Detecting and interpreting myocardial infarction using fully convolutional neural networks”. In: *Physiological Measurement* 40.1, 015001 (2019).

[92] S. W. E. Baalman et al. “A morphology based deep learning model for atrial fibrillation detection using single cycle electrocardiographic samples”. In: *International Journal of Cardiology* 316 (2020), pp. 130–136.

[93] S. Mousavi, F. Afghah, and U. R. Acharya. “HAN-ECG: An Interpretable Atrial Fibrillation Detection Model Using Hierarchical Attention Networks”. In: *Computers in Biology and Medicine* 127, 104057 (2020).

[94] D. Zhang et al. “Interpretable deep learning for automatic diagnosis of 12-lead electrocardiogram”. In: *iScience* 24.4, 102373 (2021).

[95] M. P. Witvliet et al. “Usefulness, pitfalls and interpretation of handheld single-lead electrocardiograms”. In: *Journal of Electrocardiology* 66 (2021), pp. 33–37.

[96] N. Srivastava et al. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958.

[97] X. Ying. “An Overview of Overfitting and its Solutions”. In: *Journal of Physics: Conference Series* 1168.2, 022022 (2019).

[98] M. Bodini, M. W. Rivolta, and R. Sassi. “Interpretability Analysis of Machine Learning Algorithms in the Detection of ST-Elevation Myocardial Infarction”. In: *2020 Computing in Cardiology* (13–16 Sept. 2020, Rimini, Italy). Vol. 47. New York, NY, USA: IEEE, pp. 1–4.

[99] G. Haixiang et al. “Learning from class-imbalanced data: Review of methods and applications”. In: *Expert Systems with Applications* 73.1 (2017), pp. 220–239.

[100] D. Sinnecker. “A deep neural network trained to interpret results from electrocardiograms: better than physicians?” In: *The Lancet Digital Health* 2.7 (2020), e332–e333.

[101] M. P. Turakhia et al. “Diagnostic Utility of a Novel Leadless Arrhythmia Monitoring Device”. In: *The American Journal of Cardiology* 112.4 (2013), pp. 520–524.

[102] F. N. Hatamian et al. “The effect of data augmentation on classification of atrial fibrillation in short single-lead ECG signals using deep neural networks”. In: *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (4–8 May 2020, Barcelona, Spain). New York, NY, USA: IEEE, pp. 1264–1268.

[103] Y. Liang et al. “Impact of Data Transformation: An ECG Heartbeat Classification Approach”. In: *Frontiers in Digital Health* 2, 610956 (2020).

[104] J. Gao et al. “An Effective LSTM Recurrent Network to Detect Arrhythmia on Imbalanced ECG Dataset”. In: *Journal of Healthcare Engineering* 2019, 6320651 (2019).

[105] T. Lin et al. “Focal Loss for Dense Object Detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.2 (2020), pp. 318–327.

[106] M. Buda, A. Maki, and M. A. Mazurowski. “A systematic study of the class imbalance problem in convolutional neural networks”. In: *Neural Networks* 106 (2018), pp. 249–259.

[107] C. Elkan. “The Foundations of Cost-Sensitive Learning”. In: *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI’01)* (4–10 Aug. 2021, Seattle, WA, USA). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 973–978.

[108] B. Krawczyk, M. Wozniak, and G. Schaefer. “Cost-sensitive decision tree ensembles for effective imbalanced classification”. In: *Applied Soft Computing* 14 (2014), pp. 554–562.

[109] M. Bodini, M. W. Rivolta, and R. Sassi. “Classification of 12-lead ECG With an Ensemble Machine Learning Approach”. In: *2020 Computing in Cardiology* (13–16 Sept. 2020, Rimini, Italy). Vol. 47. New York, NY, USA: IEEE, pp. 1–4.

[110] M. Bodini, M. W. Rivolta, and R. Sassi. "Classification of ECG Signals with Different Lead Systems Using AutoML". In: *2021 Computing in Cardiology* (12–15 Sept. 2021, Brno, Czech Republic). Vol. 48. New York, NY, USA: IEEE, pp. 1–4.

[111] M. Bodini, M. W. Rivolta, and R. Sassi. "Opening the black box: interpretability of machine learning algorithms in electrocardiography". In: *Philosophical Transactions of the Royal Society A* (2021), (in press).

[112] K. Thygesen et al. "Fourth Universal Definition of Myocardial Infarction (2018)". In: *European Heart Journal* 40.3 (2018), pp. 237–269.

[113] L. G. Tereshchenko and M. E. Josephson. "Frequency content and characteristics of ventricular conduction". In: *Journal of Electrocardiology* 48.6 (2015), pp. 933–937.

[114] Berkayam S. K. et al. "A survey on ECG analysis". In: *Biomedical Signal Processing and Control* 43 (2018), pp. 216–235.

[115] S. Chatterjee et al. "Review of noise removal techniques in ECG signals". In: *IET Signal Processing* 14.9 (2020), pp. 569–590.

[116] A. C. V. Maggio et al. "Quantification of Ventricular Repolarization Dispersion Using Digital Processing of the Surface ECG". In: *Advances in Electrocardiograms - Methods and Analysis*. Ed. by R. M. Millis. London, United Kingdom: IntechOpen, 2012.

[117] N. A. Pilia et al. "The impact of baseline wander removal techniques on the ST segment in simulated ischemic 12-lead ECGs". In: *Current Directions in Biomedical Engineering* 1.1 (2015), pp. 96–99.

[118] G. Lenis et al. "Comparison of Baseline Wander Removal Techniques considering the Preservation of ST Changes in the Ischemic ECG: A Simulation Study". In: *Computational and Mathematical Methods in Medicine* 2017, 9295029 (2017).

[119] L. Tan and J. Jiang. *Digital signal processing: fundamentals and applications*. 3rd ed. Cambridge, MA, United States: Academic Press, 2018.

[120] B. N. Singh and A. K. Tiwari. "Optimal selection of wavelet basis function applied to ECG signal denoising". In: *Digital Signal Processing* 16.3 (2006), pp. 275–287.

[121] M. Alfaouri and K. Daqrouq. "ECG signal denoising by wavelet transform thresholding". In: *American Journal of applied sciences* 5.3 (2008), pp. 276–281.

[122] M. A. Al-Betar. "β-Hill climbing: an exploratory local search". In: *Neural Computing and Applications* 28.1 (2016), pp. 153–168.

[123] Z. A. A. Alyasseri et al. "Hybridizing β-hill climbing with wavelet transform for denoising ECG signals". In: *Information Sciences* 429 (2018), pp. 229–246.

[124] G. Han and Z. Xu. "Electrocardiogram signal denoising based on a new improved wavelet thresholding". In: *Review of Scientific Instruments* 87.8, 084303 (2016).

[125] M. Üstündağ et al. "Denoising of weak ECG signals by using wavelet analysis and fuzzy thresholding". In: *Network Modeling Analysis in Health Informatics and Bioinformatics* 1.4 (2012), pp. 135–140.

[126] P. Singh, G. Pradhan, and S. Shahnawazuddin. "Denoising of ECG signal by non-local estimation of approximation coefficients in DWT". In: *Biocybernetics and Biomedical Engineering* 37.3 (2017), pp. 599–610.

[127] G. B. Moody and R. G. Mark. "The impact of the MIT-BIH arrhythmia database". In: *IEEE Engineering in Medicine and Biology Magazine* 20.3 (2001), pp. 45–50.

[128] N. E. Huang et al. "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis". In: *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454.1971 (1998), pp. 903–995.

[129] A. R. P. Riera et al. "The enigmatic sixth wave of the electrocardiogram: the U wave". In: *Cardiology journal* 15.5 (2008), pp. 408–421.

[130] J. Park, K. Lee, and K. Kang. "Arrhythmia detection from heartbeat using k-nearest neighbor classifier". In: *2013 IEEE International Conference on Bioinformatics and Biomedicine* (18–21 Dic. 2013, Shanghai, China). New York, NY, USA: IEEE, pp. 15–22.

[131] J. Pan and W. J. Tompkins. "A real-time QRS detection algorithm". In: *IEEE Transactions on Biomedical Engineering* BME-32.3 (1985), pp. 230–236.

[132] W. H. Jung and S. G. Lee. "An Arrhythmia Classification Method in Utilizing the Weighted KNN and the Fitness Rule". In: *IRBM* 38.3 (2017), pp. 138–148.

[133] B. Venkataramanaiah and J. Kamala. "ECG signal processing and KNN classifier-based abnormality detection by VH-doctor for remote cardiac healthcare monitoring". In: *Soft Computing* 24.22 (2020), pp. 17457–17466.

[134] W. Yang et al. "Automatic recognition of arrhythmia based on principal component analysis network and linear support vector machine". In: *Computers in Biology and Medicine* 101.1 (2018), pp. 22–32.

[135] V. Gliner and Y. Yaniv. "An SVM approach for identifying atrial fibrillation". In: *Physiological Measurement* 39.9, 094007 (2018).

[136] G. Clifford et al. "AF classification from a short single lead ECG recording: The PhysioNet/computing in cardiology challenge 2017". In: *2017 Computing in Cardiology* (24–27 Sept. 2017, Rennes, France). Vol. 44. New York, NY, USA: IEEE, pp. 1–4.

[137] C. K. Jha and M. H. Kolekar. "Cardiac arrhythmia classification using tunable Q-wavelet transform based features and support vector machine classifier". In: *Biomedical Signal Processing and Control* 59, 101875 (2020).

[138] C. Vimal and B. Sathish. "Random Forest Classifier Based ECG Arrhythmia Classification". In: *International Journal of Healthcare Information Systems and Informatics* 5.2 (2010), pp. 1–10.

[139] B. H. Kung et al. "An Efficient ECG Classification System Using Resource-Saving Architecture and Random Forest". In: *IEEE Journal of Biomedical and Health Informatics* 25.6 (2021), pp. 1904–1914.

[140] J. Rahul et al. "An improved cardiac arrhythmia classification using an RR interval-based approach". In: *Biocybernetics and Biomedical Engineering* 41.2 (2021), pp. 656–666.

[141] P. Yang et al. "Ensemble of kernel extreme learning machine based random forest classifiers for automatic heartbeat classification". In: *Biomedical Signal Processing and Control* 63, 102138 (2021).

[142] T. Ojala, M. Pietikäinen, and T. Mäenpää. "Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns". In: *Computer Vision - ECCV 2000*. Ed. by D. Vernon. Berlin, Heidelberg: Springer, pp. 404–420.

[143] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors". In: *Nature* 323.6088 (1986), pp. 533–536.

[144] G. Sannino and G. De Pietro. "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection". In: *Future Generation Computer Systems* 86 (2018), pp. 446–455.

[145] F. Bouaziz et al. "Automatic ECG arrhythmias classification scheme based on the conjoint use of the multi-layer perceptron neural network and a new improved metaheuristic approach". In: *IET Signal Processing* 13.8 (2019), pp. 726–735.

[146] J. R. Zhang et al. "A hybrid particle swarm optimization–back-propagation algorithm for feedforward neural network training". In: *Applied Mathematics and Computation* 185.2 (2007), pp. 1026–1037.

[147] R. Jothiramalingam et al. "Machine learning-based left ventricular hypertrophy detection using multi-lead ECG signal". In: *Neural Computing and Applications* 33.9 (2020), pp. 4445–4455.

[148] A. Linhart and F. Cecchi. "Common presentation of rare diseases: left ventricular hypertrophy and diastolic dysfunction". In: *International journal of cardiology* 257.15 (2018), pp. 344–350.

[149] M. Lee, T. G. Song, and J. H. Lee. "Heartbeat classification using local transform pattern feature and hybrid neural fuzzy-logic system based on self-organizing map". In: *Biomedical Signal Processing and Control* 57, 101690 (2020).

[150] A. M. Alqudah et al. "Developing of robust and high accurate ECG beat classification by combining Gaussian mixtures and wavelets features". In: *Australasian physical & engineering sciences in medicine* 42.1 (2019), pp. 149–157.

[151] M. Seera et al. "Classification of electrocardiogram and auscultatory blood pressure signals using machine learning models". In: *Expert Systems with Applications* 42.7 (2015), pp. 3643–3652.

[152] R. G. Afkhami, G. Azarnia, and M. A. Tinati. "Cardiac arrhythmia classification using statistical and mixture modeling features of ECG signals". In: *Pattern Recognition Letters* 70.15 (2016), pp. 45–51.

[153] D. Lai et al. "Non-Standardized Patch-Based ECG Lead Together With Deep Learning Based Algorithm for Automatic Screening of Atrial Fibrillation". In: *IEEE Journal of Biomedical and Health Informatics* 24.6 (2020), pp. 1569–1578.

[154] T. F. Romdhane et al. "Electrocardiogram heartbeat classification based on a deep convolutional neural network and focal loss". In: *Computers in Biology and Medicine* 123, 103866 (2020).

[155] S. Ioffe and C. Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *Proceedings of the 32nd International Conference on Machine Learning* (6–11 July 2015, Lille, France). Ed. by F. Bach and D. Blei. Vol. 37. Atlanta, GA, USA: PMLR, pp. 448–456.

[156] M. Degirmenci et al. "Arrhythmic Heartbeat Classification Using 2D Convolutional Neural Networks". In: *IRBM* (2021), in press.

[157] I. Silva and G. B. Moody. "An Open-source Toolbox for Analysing and Processing PhysioNet Databases in MATLAB and Octave". In: *Journal of Open Research Software* 2.1, e27 (2014).

[158] Y. LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[159] H. I. Fawaz et al. "Deep learning for time series classification: a review". In: *Data Mining and Knowledge Discovery* 33.4 (2019), pp. 917–963.

[160] S. Hochreiter and J. Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (1997), pp. 1735–1780.

[161] L. Sun et al. "A stacked LSTM for atrial fibrillation prediction based on multivariate ECGs". In: *Health Information Science and Systems* 8.1, 19 (2020).

[162] S. Petrutiu, A. V. Sahakian, and S. Swiryn. "Abrupt changes in fibrillatory wave characteristics at the termination of paroxysmal atrial fibrillation in humans". In: *EP Europace* 9.7 (2007), pp. 466–470.

[163] G. E. Moody. "Spontaneous termination of atrial fibrillation: a challenge from physionet and computers in cardiology 2004". In: *Computers in Cardiology, 2004* (19–22 Sept. 2004, Chicago, IL, USA). Ed. by A. Murray. Vol. 31. New York, NY, USA: IEEE, pp. 101–104.

[164] G. Petmezas et al. "Automated Atrial Fibrillation Detection using a Hybrid CNN-LSTM Network on Imbalanced ECG Datasets". In: *Biomedical Signal Processing and Control* 63, 102194 (2021).

[165] G. Moody. "A new method for detecting atrial fibrillation using R-R intervals". In: *Computers in Cardiology, 1983* (21–24 Sept. 1983, Thessaloniki, Greece). Ed. by A. Murray. New York, NY, USA: IEEE, 1983, pp. 227–230.

[166] C. Chen et al. "Automated arrhythmia classification based on a combination network of CNN and LSTM". In: *Biomedical Signal Processing and Control* 57, 101819 (2020).

[167] H. Sohn, K. Worden, and C. Farrar. "Novelty detection using auto-associative neural network". In: *Proceedings of 2001 ASME International Mechanical Engineering Congress and Exposition* (11–16 Nov. 2001, New York, NY, USA). Los Alamos, NM, USA: Los Alamos National Laboratory.

[168] Matteo Bodini. "Aspect Extraction from Bangla Reviews Through Stacked Auto-Encoders". In: *Data* 4.3, 121 (2019).

[169] B. Hou et al. "LSTM-Based Auto-Encoder Model for ECG Arrhythmias Classification". In: *IEEE Transactions on Instrumentation and Measurement* 69.4 (2020), pp. 1232–1240.

[170] S. Nurmaini et al. "Deep Learning-Based Stacked Denoising and Autoencoder for ECG Heartbeat Classification". In: *Electronics* 9.1, 135 (2020).

[171] R. Siouda, M. Nemissi, and H. Seridi. "ECG beat classification using neural classifier based on deep autoencoder and decomposition techniques". In: *Progress in Artificial Intelligence* 10.3 (2021), pp. 333–347.

[172] N. V. Chawla et al. "SMOTE: Synthetic Minority Over-sampling Technique". In: *Journal of Artificial Intelligence Research* 16.1 (2002), pp. 321–357.

[173] J. H. Friedman. "Greedy function approximation: a gradient boosting machine". In: *The Annals of Statistics* 29.5 (2001), pp. 1189–1232.

[174] J. Friedman, T. Hastie, and R. Tibshirani. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. New York, NY, USA: Springer, 2009.

[175] B. M. Greenwell. "pdp: An R Package for Constructing Partial Dependence Plots". In: *The R Journal* 9.1 (2017), pp. 421–436.

[176] D. E. Matsumoto, ed. *The Cambridge dictionary of psychology*. Cambridge, United Kingdom: Cambridge University Press, 2009.

[177] S. Wachter, B. Mittelstadt, and C. Russell. "Counterfactual explanations without opening the black box: automated decisions and the GDPR". In: *Harvard Journal of Law & Technology* 31.2 (2018), pp. 841–887.

[178] S. Craw. "Manhattan Distance". In: *Encyclopedia of Machine Learning and Data Mining*. Ed. by C. Sammut and G. I. Webb. Boston, MA, USA: Springer, 2017, pp. 790–791.

[179] Z. Cui et al. "Optimal Action Extraction for Random Forests and Boosted Trees". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (10–13 Aug. 2015, Sydney, NSW, Australia). New York, NY, USA: ACM, pp. 179–188.

[180] R. Poyiadzi et al. "FACE: Feasible and Actionable Counterfactual Explanations". In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, AIES '20* (7–9 Feb. 2020, New York, NY, USA). New York, NY, USA: ACM, pp. 344–350.

[181] H. Geoffrey and F. Nicholas. "Distilling a Neural Network Into a Soft Decision Tree". In: *1st International Workshop on Comprehensibility and Explanation in AI and ML, CEX 2017* (16–17 Nov. 2017, Bari, Italy). Vol. 2071. Aachen, Germany: CEUR Workshop Proceedings, pp. 451–454.

[182] C. Yang, A. Rangarajan, and S. Ranka. "Global Model Interpretation Via Recursive Partitioning". In: *2018 IEEE 20th International Conference on High Performance Computing and Communications* (28–30 June 2018, Exeter, United Kingdom). New York, NY, USA: IEEE, pp. 1563–1570.

[183] M. T. Ribeiro, S. Singh, and C. Guestrin. ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (13–17 Aug. 2016, San Francisco, CA, USA). New York, NY, USA: ACM, pp. 1135–1144.

[184] E. Amparore, A. Perotti, and P. Bajardi. "To trust or not to trust an explanation: using LEAF to evaluate local linear XAI methods". In: *PeerJ Computer Science* 7, e479 (2021).

[185] S. D. Goodfellow et al. "Towards Understanding ECG Rhythm Classification Using Convolutional Neural Networks and Attention Mappings". In: *Proceedings of the 3rd Machine Learning for Healthcare Conference* (17–18 Aug. 2018, Palo Alto, CA, USA). Ed. by F. Doshi-Velez et al. Vol. 85. Atlanta, GA, USA: PMLR, pp. 83–101.

[186] B. Zhou et al. “Learning Deep Features for Discriminative Localization”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (27–30 June 2018, Las Vegas, NV, USA). New York, NY, USA: IEEE, pp. 2921–2929.

[187] S. A. Hicks et al. “Explaining deep neural networks for knowledge discovery in electrocardiogram analysis”. In: *Scientific Reports* 11.1, 10949 (2021).

[188] R. R. Selvaraju et al. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization”. In: *2017 IEEE International Conference on Computer Vision (ICCV)* (22–29 Oct. 2017, Venice, Italy). New York, NY, USA: IEEE, pp. 618–626.

[189] C. R. Juhl et al. “Hidradenitis suppurativa and electrocardiographic changes: a cross-sectional population study”. In: *British Journal of Dermatology* 178.1 (2018), pp. 222–228.

[190] S. Vijayarangan et al. “Interpreting Deep Neural Networks for Single-Lead ECG Arrhythmia Classification”. In: *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)* (20–24 July 2020, Montreal, QC, Canada). New York, NY, USA: IEEE, pp. 300–303.

[191] Q. Yao et al. “Multi-class Arrhythmia detection from 12-lead varied-length ECG using Attention-based Time-Incremental Convolutional Neural Network”. In: *Information Fusion* 53 (2020), pp. 174–182.

[192] F. Liu et al. “An open access database for evaluating the algorithms of electrocardiogram rhythm and morphology abnormality detection”. In: *Journal of Medical Imaging and Health Informatics* 8.7 (2018), pp. 1368–1373.

[193] M. Schuster and K. K. Paliwal. “Bidirectional recurrent neural networks”. In: *IEEE Transactions on Signal Processing* 45.11 (1997), pp. 2673–2681.

[194] S. Hong et al. “MINA: Multilevel Knowledge-Guided Attention for Modeling Electrocardiography Signals”. In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)* (10–16 Aug. 2019, Macao, China). Ed. by S. Kraus. San Mateo, CA, USA: IJCAI, pp. 5888–5894.

[195] E. Shelhamer, J. Long, and T. Darrell. “Fully Convolutional Networks for Semantic Segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.4 (2017), pp. 640–651.

[196] A. Shrikumar, P. Greenside, and A. Kundaje. “Learning important features through propagating activation differences”. In: *Proceedings of the 34th International Conference on Machine Learning* (6–11 Aug. 2017, Sydney, NSW, Australia). Ed. by D. Precup and Y. W. Teh. Vol. 70. PMLR. Atlanta, GA, USA, pp. 3145–3153.

[197] H. He and Y. Tan. “Automatic pattern recognition of ECG signals using entropy-based adaptive dimensionality reduction and clustering”. In: *Applied Soft Computing* 55 (2017), pp. 238–252.

[198] S. M. Lundberg and S. I. Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems* (4–9 Dic. 2017, Long Beach, CA, USA). Ed. by U. Von Luxburg et al. Red Hook, NY, USA: Curran Associates Inc., pp. 4765–4774.

[199] L. S. Shapley. "Notes on the $n-$Person Game — II: The value of an $n-$Person Game". In: *Research Memorandum* RM-670 (1951).

[200] S. Moretti and F. Patrone. "Transversality of the Shapley value". In: *TOP* 16, 1 (2008).

[201] L. Ibrahim et al. "Explainable Prediction of Acute Myocardial Infarction Using Machine Learning and Shapley Values". In: *IEEE Access* 8 (2020), pp. 210410–210417.

[202] C. D. Woody. "Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals". In: *Medical and biological engineering* 5.6 (1967), pp. 539–554.

[203] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv:1412.6980* (2014).

[204] X. Zhou et al. "Premature Ventricular Contraction Detection from Ambulatory ECG Using Recurrent Neural Networks". In: *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (18–21 July 2018, Honolulu, HI, USA). New York, NY, USA: IEEE, pp. 2551–2554.

[205] F. Badilini et al. "QT interval analysis on ambulatory electrocardiogram recordings: a selective beat averaging approach". In: *Medical & Biological Engineering & Computing* 37.1 (1999), pp. 71–79.

[206] J. Zheng et al. "Optimal Multi-Stage Arrhythmia Classification Approach". In: *Scientific Reports* 10.1, 2898 (2020).

[207] M. Zöller and M. F. Huber. "Benchmark and Survey of Automated Machine Learning Frameworks". In: *Journal of Artificial Intelligence Research* 70 (2021), pp. 409–472.

[208] J. Waring, C. Lindvall, and R. Umeton. "Automated machine learning: Review of the state-of-the-art and opportunities for healthcare". In: *Artificial Intelligence in Medicine* 104, 101822 (2020).

[209] J. K. Chow et al. "Anomaly detection of defects on concrete structures with the convolutional autoencoder". In: *Advanced Engineering Informatics* 45, 101105 (2020).

[210] L. Bottou et al. "Scaling Learning Algorithms toward AI". In: *Large-Scale Kernel Machines*. New York, NY, USA: IEEE, 2007, pp. 321–359.

[211] A. L. Blum and P. Langley. "Selection of relevant features and examples in machine learning". In: *Artificial Intelligence* 97.1-2 (1997), pp. 245–271.

[212] G. Chandrashekar and F. Sahin. "A survey on feature selection methods". In: *Computers & Electrical Engineering* 40.1 (2014), pp. 16–28.

[213] P. Pudil, J. Novovičová, and J. Kittler. "Floating search methods in feature selection". In: *Pattern Recognition Letters* 15.11 (1994), pp. 1119–1125.

[214] M. Feurer et al. "Efficient and Robust Automated Machine Learning". In: *Proceedings of the 28th International Conference on Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 2. Cambridge, MA, USA: MIT Press, 2015, pp. 2962–2970.

[215] H. Jin, Q. Song, and X. Hu. "Auto-Keras: An Efficient Neural Architecture Search System". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (4–8 Aug. 2019, Anchorage, AK, USA). New York, NY, USA: ACM, pp. 1946–1956.

[216] T. T. Le, W. Fu, and J. H. Moore. "Scaling tree-based automated machine learning to biomedical big data with a feature set selector". In: *Bioinformatics* 36.1 (2020), pp. 250–256.

[217] J. T. Schaffer. "How should the ECG be Used in the Chest Pain Patient?" In: *Critical Decisions in Emergency and Acute Care Electrocardiography*. Ed. by W. J. Brady and J. D. Truwit. Hoboken, NJ, United States: Wiley-Blackwell, 2009, pp. 49–57.

[218] A. K. Dohare, V. Kumar, and R. Kumar. "Detection of myocardial infarction in 12 lead ECG using support vector machine". In: *Applied Soft Computing* 64 (2018), pp. 138–147.

[219] M. P. Than et al. "Machine Learning to Predict the Likelihood of Acute Myocardial Infarction". In: *Circulation* 140.11 (2019), pp. 899–909.

[220] U. B. Baloglu et al. "Classification of myocardial infarction with multi-lead ECG signals and deep CNN". In: *Pattern Recognition Letters* 122.1 (2019), pp. 23–30.

[221] W. Liu et al. "MFB-CBRNN: A Hybrid Network for MI Detection Using 12-Lead ECGs". In: *IEEE Journal of Biomedical and Health Informatics* 24.2 (2020), pp. 503–514.

[222] Z. He et al. "MFB-LANN: A lightweight and updatable myocardial infarction diagnosis system based on convolutional neural networks and active learning". In: *Computer Methods and Programs in Biomedicine* 210, 106379 (2021).

[223] M. W. Rivolta, L. T. Mainardi, and R. Sassi. "Quantification of ventricular repolarization heterogeneity during moxifloxacin or sotalol administration using-index". In: *Physiological measurement* 36.4, 803 (2015).

[224] D. Levy et al. "Electrocardiographic changes with advancing age. A cross-sectional study of the association of age with QRS axis, duration and voltage". In: *Journal of Electrocardiology* 20 (1987), pp. 44–47.

[225] D. Alvarez-Melis and T. S. Jaakkola. "On the robustness of interpretability methods". In: *arXiv:1806.080* (2018).

[226] S. Minaee et al. "Image Segmentation Using Deep Learning: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021), in press.

[227] M. D. Zeiler and R. Fergus. "Visualizing and Understanding Convolutional Networks". In: *Computer Vision – ECCV 2014* (6–12 Sept. 2014, Zurich, Switzerland). Ed. by D. Fleet et al. Cham, Switzerland: Springer, pp. 818–833.

[228] K. Simonyan, A. Vedaldi, and A. Zisserman. "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps". In: *2nd International Conference on Learning Representations, ICLR 2014 - Workshop Track Proceedings* (14–16 Apr. 2014, Banff, AB, Canada). La Jolla, CA, USA: ICLR, pp. 1–8.

[229] T. Vicar et al. “ECG Abnormalities Recognition Using Convolutional Network with Global Skip Connections and Custom Loss Function”. In: *2020 Computing in Cardiology* (13–16 Sept. 2020, Rimini, Italy). Vol. 47. New York, NY, USA: IEEE, pp. 1–4.

[230] E. Hellinger. “Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen.” In: *Journal für die reine und angewandte Mathematik* 1909.136 (1909), pp. 210–271.

[231] Z. Q. Lin et al. “Do explanations reflect decisions? a machine-centric strategy to quantify the performance of explainability algorithms”. In: *arXiv:1910.07387* (2019).

[232] F. Liu et al. “Performance Analysis of Ten Common QRS Detectors on Different ECG Application Cases”. In: *Journal of Healthcare Engineering* 2018, 9050812 (2018).

List of Publications

[1] M. Bodini, M. W. Rivolta, and R. Sassi. “Classification of 12-lead ECG With an Ensemble Machine Learning Approach”. In: 2020 Computing in Cardiology (13–16 Sept. 2020, Rimini, Italy). Vol. 47. New York, NY, USA: IEEE, pp. 1–4.

[2] M. Bodini, M. W. Rivolta, and R. Sassi. “Classification of ECG Signals with Different Lead Systems Using AutoML”. In: 2021 Computing in Cardiology (12–15 Sept. 2021, Brno, Czech Republic). Vol. 48. New York, NY, USA: IEEE, pp. 1–4.

[3] M. Bodini, M. W. Rivolta, and R. Sassi. “Interpretability Analysis of Machine Learning Algorithms in the Detection of ST-Elevation Myocardial Infarction”. In: 2020 Computing in Cardiology (13–16 Sept. 2020, Rimini, Italy). Vol. 47. New York, NY, USA: IEEE, pp. 1–4.

[4] M. Bodini, M. W. Rivolta, and R. Sassi. “Opening the black box: interpretability of machine learning algorithms in electrocardiography”. In: Philosophical Transactions of the Royal Society A (2021), (in press).