Expert Systems With Applications 238 (2024) 121772

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An intelligent Medical Cyber–Physical System to support heart valve disease screening and diagnosis

Gennaro Tartarisco [a,*,1], Giovanni Cicceri [b,1], Roberta Bruschetta [c], Alessandro Tonacci [d], Simona Campisi [c], Salvatore Vitabile [b], Antonio Cerasa [a,g,h], Salvatore Distefano [e,2], Alessio Pellegrino [f], Pietro Amedeo Modesti [f,2], Giovanni Pioggia [a,2]

[a] Institute for Biomedical Research and Innovation, National Research Council, IRIB-CNR, Messina, Italy
[b] Department of Biomedicine, Neuroscience and Advanced Diagnostics (BiND), University of Palermo, Palermo, Italy
[c] Department of Engineering, Campus Bio-Medico University, Rome, Italy
[d] Clinical Physiology Institute, National Research Council of Italy (IFC-CNR), Pisa, Italy
[e] MIFT Department, University of Messina, Messina, Italy
[f] Department of Experimental and Clinical Medicine, University of Florence, Florence, Italy
[g] Pharmacotechnology Documentation and Transfer Unit, Preclinical and Translational Pharmacology, Department of Pharmacy, Health Science and Nutrition, University of Calabria, Arcavacata, Italy
[h] S. Anna Institute, Crotone, Italy

ARTICLE INFO

Keywords:
Heart sound signal
Heart disease classification
Machine learning
Cyber–physical systems
Cloud computing

ABSTRACT

Cardiovascular diseases are currently the major causes of death globally. Among the strategies to prevent cardiovascular issues, the automated classification of heart sound abnormalities is an efficient way to detect early signs of cardiac conditions leading to heart failure or other, even asymptomatic, complications, quite effective for timely interventions. Despite the significant improvements in this field, there are still limitations due to the lack of solutions, available data-sets and poor (mainly binary — normal vs abnormal) classification models and algorithms. This paper presents a Medical Cyber–Physical System (MCPS) for the automatic classification of heart valve diseases onsite, in a timely manner. The proposed MCPS, indeed, can be deployed into personal and mobile devices, addressing the limitations of existing solutions for patients, healthcare practitioners, and researchers, through an efficient and easy accessible tool. It combines different neural network models trained on a new Italian dataset of 132 adult patients covering 9 heart sound categories (1 normal and 8 abnormal), also validated against two main open-access (Physionet/CinC Challenge 2016 and Korean) datasets. The overall MCPS performance (time, processing and energy resource utilization) and the high accuracy of the models (up to 98%) demonstrated the feasibility of the proposed solution, even with few data. The dataset supporting the findings of this paper is available upon request to the authors.

1. Introduction

Cardiovascular diseases are the main cause of global mortality with an estimated number of deaths which steadily on the rise from 12.1 million in 1990 to 18.6 million in 2019 (32% of all deaths) (Roth et al., 2020). They are also the most frequent cause of hospitalization in people aged over 65, with costs expected to reach around $1.1 trillion in the U.S. by 2035 (Association et al., 2017). In this scenario, early recognition of heart valve diseases (HVD) by, e.g., processing cardiac sounds, can be effective in preventing and even improving disease management. Despite sophisticated medical technologies like ultrasound imaging and Eco-Doppler, cardiac auscultation is still the primary tool used by professionals.

In order for the medical staff to acquire auscultation skills in assessing and diagnosing HVD, practice and training with experienced mentors and a high number of patients (Chizner, 2008) is required. In the last few years, the advances in digital signal processing have

* Corresponding author.
E-mail addresses: gennaro.tartarisco@irib.cnr.it (G. Tartarisco), giovanni.cicceri@unipa.it (G. Cicceri), roberta.bruschetta@irib.cnr.it (R. Bruschetta), alessandro.tonacci@cnr.it (A. Tonacci), simona.campisi@irib.cnr.it (S. Campisi), salvatore.vitabile@unipa.it (S. Vitabile), antonio.cerasa@irib.cnr.it (A. Cerasa), salvatore.distefano@unime.it (S. Distefano), alessio.pellegrino@unifi.it (A. Pellegrino), pa.modesti@unifi.it (P.A. Modesti), giovanni.pioggia@cnr.it (G. Pioggia).
[1] These authors share the first authorship.
[2] These authors share the last authorship.

https://doi.org/10.1016/j.eswa.2023.121772
Received 30 July 2023; Received in revised form 30 August 2023; Accepted 20 September 2023
Available online 28 September 2023

focused on analysing acoustic cardiac signals combined with intelligent algorithms for automatic classification of heart murmurs (Bhandare, Patel, & Shimshak, 2022; Sethi et al., 2022). Many pathological conditions that cause murmurs and aberrations of heart sounds manifest much earlier in phonocardiography than they are reflected by symptoms (Shanthi, Anand, Annapoorani, & Birundha, 2023). Thus, corrective measures can be taken by properly interpreting the phonocardiogram (PCG) signal.

In most cases, the activities in the PCG signal related to a given disease are contained in a single interval of cardiac cycles. Several efforts have provided significant research about the automatic classification of heart valve disease to date, mainly in terms of accuracy of classification between normal vs. abnormal class (Dwivedi, Imtiaz, & Rodriguez-Villegas, 2018). Former studies started with machine learning (ML) models and features extracted from time/frequency domain (e.g. energy, entropy, Fourier transformation) to multi-resolution analysis (e.g. wavelet parameters). Recent works adopted deep learning (DL) models able to extract a high-level of information without any human hand-crafted features as proposed by Clifford et al. (2017). Overall, as outlined by Dong et al. (2019), there is a trend to classify the whole audio signal without any segmentation step.

Despite such progresses, there is still a lack of heart sound data and datasets to develop intelligent solutions able to recognize pathological heart issues. This approach could play an important role in terms of use and cost-effectiveness, making intelligent phonocardiogram-based support tools available to every physician to reduce the referral of patients to poorly affordable and expensive tests. Even better if such solutions and tools can be exploited pervasively and ubiquitously anywhere and from everywhere, thus allowing to support the diagnosis in a timely manner and promptly reacting in the case of anomalies to prevent heart issues and failures.

Today, there are digital stethoscopes able to record, store, and visualize heart sound signals, but the research and application of intelligent auscultation algorithms is still poorly adequate. Advances of remote computer-assisted auscultation systems will improve the ability of healthcare workers to screen and diagnose early heart symptoms and reduce patient contact preventing transmission, especially during the COVID-19 pandemic (Vasudevan et al., 2020). The need and impact of such tools would be much greater in low income countries, where a shortage of healthcare resources is present (skilled doctors cannot be dedicated to screening purposes) and the prevalence of rheumatic heart disease (the most common cause of primary valvular heart disease Coffey et al., 2021) remains high, specially in Oceania (where the age-standardized mortality is highest), Africa and Asia (Watkins et al., 2017). Therefore, automated systems might be particularly useful in least developed countries.

In this context, this paper proposes a full fledged, automatic HVD diagnosis (HVDD) Medical Cyber–Physical System (MCPS), aiming to design and assess the feasibility and effectiveness of the proposed HVDD MCPS solution. Thereby, this study focuses on three main research questions: (i) how to capture and ingest patient data? (Physical Layer and HVDD device); (ii) how to manage and process collected data (sound) to achieve a high accuracy in the detailed, multiclass (1 normal + 8 abnormal classes) diagnosis of heart valve disease risk? (Cyber Layer and HVDDaaS classifier); and (iii) is the HVDD system client suitable for deployment on a resource-constrained (mobile, personal) device in real-time? (CPU, memory, storage, network, energy resource utilization, latency and reliability analyses).

The main contribution of this paper is 5-fold: (i) a Medical Cyber–Physical System framing the audio sensors and their data processing components altogether into a solution deployable in and exploitable by personal and mobile devices in real time; (ii) the HVD dataset repository[3] including 132 patients labelled with 9 different normal and abnormal (either aortic or mitral, stenosis or regurgitation, moderate or severe) heart valve conditions; (iii) an intelligent framework hierarchically combining different machine learning models by voting mechanisms to overcome the lack of data issues; (iv) the assessment of the full HVDD framework in terms of model accuracy (up to 98%) and feasibility/resource utilization, further validating the models on two open-access datasets, Physionet/CinC Challenge 2016 and a Korean one; (v) model comparison and guidelines for select the proper structure based on the available data.

The rest of this paper is organized as follows: in Section 2, the approach methodologies and techniques adopted for the proposed solution, including the design of the HVDD MCPS and the HVD dataset, are discussed. Section 3 details the HVD classifier models, then Section 4 reports on their training, test and validation stages. Section 5 reports the experimental results obtained by validating the HVD models against the HVD dataset, also compared against two (Physionet/CinC Challenge 2016 and Korean) public datasets. Section 6 provides further insights on the HVDD MCPS feasibility and model selection guidelines, while conclusions and future work are discussed in Section 7.

2. Material and methods

In this section, we provide a comprehensive overview of the MCPS and HVDD system, outlining the workflow and processes used. The Italian Heart Sound Dataset, consisting of data from 132 subjects, is also introduced and discussed in relation to the data pre-processing and feature extraction module, which prepares and extracts relevant information for the HVD machine learning classifier in Section 3.

2.1. The Medical Cyber–Physical System (MCPS)

The approach proposed in this paper aims at automatically diagnosing heart valve diseases starting from the patient heart sounds. To such a purpose, an MCPS starting from an automatic tool for screening and diagnosis of heart valve disease is proposed. It is based on ML models running on a (personal and/or mobile) device (e.g., laptop, tablet or even smartphone) equipped or connected with a digital stethoscope for auscultation (or, ultimately, even using the device mic), to provide real-time diagnosis or even remotely, supported by a Cloud-based HVDD service.

Specifically, the architecture of the proposed system for the automatic diagnosis of heart valve diseases (HVDD) includes both the physical part, i.e. the patient, and the digital/cyber part, devoted to the digitalization of the patient HVD condition information by, e.g., a digital stethoscope probing the patient heart, and to the management and processing of collected data, thus implementing an MCPS. Based on the device physical resource availability, including the sensing and network facilities, such an HVDD MCPS has to be implemented and provisioned in different *"flavours"*. In general terms, the basic idea is to implement a ubiquitous service continuously updating the original intelligent HVDD system with new patient data (from the physical layer to the cyber layer) and up to date models (conversely). Therefore, it is required to manage the dataset, gathering new data from multiple digital stethoscopes and device sampling patients, to train and tune the ML model, following a continuous learning approach.

Such requirements push towards a Cloud-based architecture for implementing the HVDD MCPS cyber layer, consisting of a model of the physical system (the ML model as a sort of digital twin of the heart valves) and the patient data collections. Thus, it provides services to the underlying physical system to manage historical patient data (anonymization, curation, filtering, cleansing, aggregation, feature engineering, storage) and to continuously train and tune the ML model for improving it, implementing the HVDD as a Service (HVDDaaS). On the other hand, at the physical layer of the HVDD MCPS, a device has to interact with the HVDDaaS in different ways, based on its resource capabilities. Two ways of interaction with the HVDDaaS are possible: offline and online.

[3] Upon request.

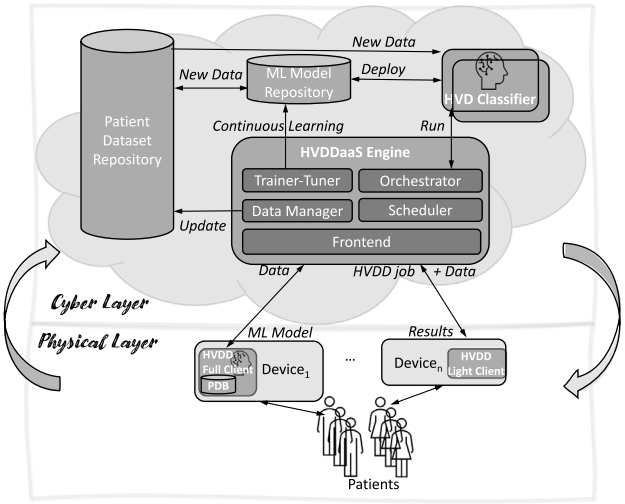


Fig. 1. HVDD Architecture.

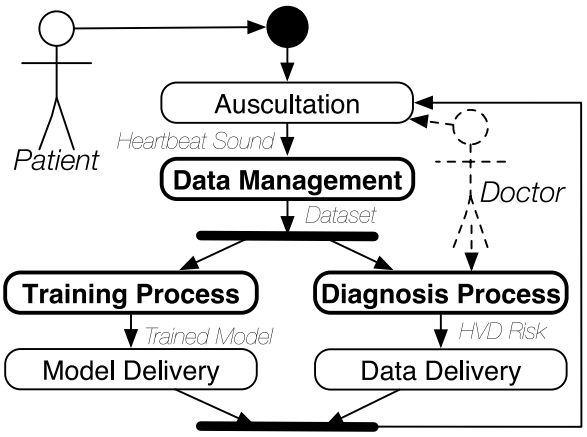


Fig. 2. The HVDD overall workflow.

- The *offline mode* is mainly suitable for devices with no or unstable network connections, and requires that a device is powerful enough to run the pre-trained model previously downloaded from the HVDDaaS system and installed on-board. Thereby, the patient data are collected and immediately processed locally by the HVD Classifier running on the device.
- On the other hand, the *online mode* works online, thus only requiring a fully operating network connection. The patient data are only captured by the device and immediately sent to the HVDDaaS Engine for processing, on-demand, as a service. Thus, the HVDDaaS Engine components process the patient data by a specific HVD Classifier deployed and running on a virtual machine or a container instance on the Cloud. It could be implemented as a Web service, thus requiring a light client, even just a Web browser, allowing to run on almost any computing system without installing any external software.

In both cases, all devices have to send patient data to the Cloud server for continuous learning, and for training and tuning the ML models. In the offline mode, such a data transfer would take place when the network connection would be available, thus requiring a specific local Patient DB (PDB) as a temporary buffer for patient data.

2.1.1. HVDD MCPS architecture

In light of this, the reference architecture of Fig. 1 identifies the main components of the HVDD framework. Overall, it can be considered as a client–server architecture, with two different clients: the offline and the online ones. On the server side, the core component is represented by the HVDDaaS Engine, managing the server system and providing the main services to the clients through its *Frontend*. The HVDDaaS Engine *Data Manager* manages data, collecting, filtering, cleaning, aggregating and anonymizing them, also applying feature engineering processes to be ready for training/tuning operations on the ML model. Such activities are coordinated by the *Trainer–Tuner* component of the Engine, in turn interacting with both the *Patient Dataset* and the *ML Model Repository*, storing the patient datasets and the ML models, respectively. The Engine *Scheduler* and *Orchestrator* manage the online client requests for HVDD task processing, scheduling such incoming requests according to specific policies (round robin, priority, etc.), and then instantiating (upon availability), deploying and orchestrating HVD Classifier nodes, respectively.

The HVDD client can be logically distinguished into two modes: offline and online. They are implemented by two different clients, a lightweight one, the *HVDD Light Client*, and the *HVDD Full Client*. The former is mainly tasked at working in the online mode only, and is conceived for being executed in resource constrained devices that are not able to run the full client or choose to operate always online. Overall, it is just composed of a lightweight *HVDD Client Frontend* that mainly interacts with the user, then collects and sends the heart sound data to the HVDDaaS Frontend, receiving back and showing the processing results. On the other hand, the *HVDD Full Client* can operate both offline and online just selecting the related mode option once captured the heart sound data. In the online mode, it acts as a light client. In the offline mode, it processes the data and then runs the local HVD Classifier for inferring a proper diagnosis. All such data, including the feature extracted from heart sounds, are collected and stored locally on the patient dataset repository (*PDS*), and sent to the HVDDaaS Engine when the device is online. When online it also periodically checks, by querying the HVDDaaS Engine, about ML model updates.

2.1.2. HVDD behaviour

To better understand how the proposed HVDD system works, the workflow depicted in Fig. 2 is introduced by an UML activity diagram (AD) notation. It describes, from a high-level standpoint, the full HVDD process, without taking into account the operating mode (online–offline) that can slightly change the workflow logic but not its main three steps: Data Management, Training and Diagnosis (thus highlighted in the figure).

It is triggered by the patient or the caregiver aiming to assess the HVD risk of the former. To such a purpose, the workflow starts by listening to the heartbeat of the patient (*Auscultation*) with a digital stethoscope, either the smartphone microphone or more professional digital stethoscopes, as previously outlined. The heartbeat sound captured is then managed (filtered, stored and preprocessed) in the *Data Management* step, making the dataset ready for processing. Thereby, the process forks (represented by the AD thick fork bar) into two parallel activities: training and classification. In the *Training Process*, the dataset is delivered to the Cloud for training or tuning (in the case of incremental-continuous learning) the ML models for the HVD classifiers. The *Diagnosis Process*, performed either remotely by a specific HVD classifier deployed on the Cloud in the case of online mode, or locally on the device in the case of the offline mode, provides a report quantifying the *HVD Risk*. This should involve a doctor, if present, who can acknowledge the report or further investigate about the issue (if any). At the end of training and diagnosis phases, the corresponding repositories are updated with the new HVD trained model (*Model Delivery*) and the patient data (*Data Delivery*), respectively, thus ready to restart with a new auscultation (after the AD join connector bar).

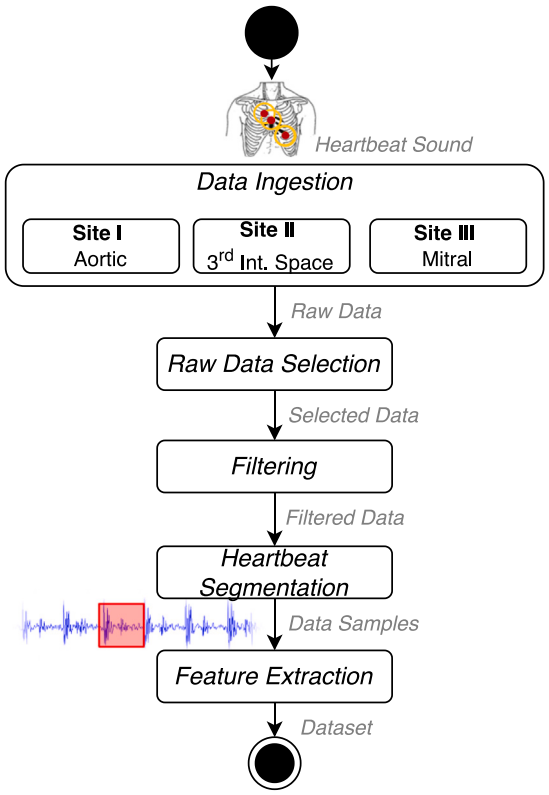


Fig. 3. The HVDD data management workflow.

2.2. Data management

The first stage of the HVDD workflow, after auscultation, focuses on data management. It is a complex process aiming to transform the heartbeat sound into a (machine readable/processable) dataset, articulated into the different activities shown in Fig. 3. More specifically, the heartbeat sound is first captured from 3 auscultation sites (*Data Ingestion*) and the raw data thus obtained are then selected (*Raw Data Selection*), filtered (*Filtering*) and sampled (*Heartbeat Segmentation*). The data sample is finally processed by a feature engineering algorithm to extract features and to identify the HVD dataset (*Feature Extraction*), ready to be further elaborated in the next stages. Details about all such steps are reported in the following.

2.2.1. Dataset

The study presented here was conducted in collaboration with the cardiologists of the Institute of Clinical Physiology of Pisa and with the Department of Medical and Surgical Critical Care, University of Florence. The study was approved by the Ethical Committee Area Vasta Centro of the Azienda Ospedaliero-Universitaria Careggi, Florence, Italy (Ref. OSS.14.089).

The study included 132 subjects (females: 53, males: 79), among which 38 healthy individuals and 94 patients with valvular disease. All participants and/or parents gave written, informed consent to participate in this study implying storage and processing of their personal data. The age of participants ranged from 41 to 65 years (50 ±10 years). The heart sound audio was collected using the Texas Instruments digital stethoscope and the medical development kit (MDK) based on the digital signal processor (DSP) TMS320C5515 (Markandey, 2010), which consists of three main components: the digital signal processing unit, the front board and the listening sensor. Such a tool employs an analogic front end to capture the acoustic sound waves of the heart. The analogic signals are amplified and digitalized before being transmitted to the processor for further processing and data transmission via the Bluetooth interface.

Heart sounds have been recorded from three auscultation sites: aortic (A), third intercostal space (T) and mitral (M) in supine position, as shown in Fig. 3. The pulmonic area of auscultation has been excluded since it is very close and strongly correlated with the aortic one. For each patient, three separated heart sound recordings ranging from approximately 90 s to more than 270 s have been captured. In Table 1, the number of subjects for each pathology (label) collected and validated by the gold standard Echocardiography is reported. Specifically, the Table reports the *cardinality* of the dataset, i.e., the number of subjects in the *Train&Test*, *Validation* and *Overall* datasets of the corresponding row label. The *Classification Tasks* column refers to the (binary) categorization (see Section 3.3) based on the *status* (healthy-diseased), *valve* (aortic-mitral), *cause* (stenosis-regurgitation), and *degree* (moderate-severe).

This dataset brings together a relevant number of sounds on a large cohort of patients (132). It could be a relevant starting point for detecting hearth failures and anomalies, but the real challenge is to provide a detailed classification. Despite in such cases it could somehow results rather sparse, and imbalanced, (DMMo, DAStMo, DMStSe, DMReMO only have one sample, no subject in the DMStMo one), to the best of our knowledge this is the first dataset with a detailed labelling to allow such deep investigations (multiclass on HVD).

2.2.2. Data preprocessing and feature extraction

The data preprocessing process is a crucial step in preparing the audio files for input into neural networks. It includes several key components to ensure that the data is cleaned, organized, and ready for analysis. The first step of preprocessing involves removing unusable audio files that are either corrupted or do not contain relevant information. Next, frequency noise is removed by filtering raw heartbeat signals and extracting informative bandwidths. To account for the logarithmic nature of human perception of sound intensity (Varshney & Sun, 2013), we use a logarithmic scale and Short-time Fourier transform (STFT) filter to remove noise from signals within the frequency bands of 1 to 40 Hz for low frequency and above 512 Hz for high frequency using the librosa Python package (McFee et al., 2015). To ensure recognition of potential cardiac irregularities, we use the first 60 s of all heartbeat records at a sample rate of 11 025 Hz. Once the audio signal is cleaned up, it is segmented into 30 segments of 2 s each and features are extracted from each segment. Specifically, we use the Mel Frequency Cepstral Coefficients (MFCC) method (El Badlaoui & Hammouch, 2017), which is a successful feature extracted for the analysis of audio files and heartbeats (Rubin et al., 2016). MFCC captures and compacts spectral characteristics and time variations (Usman, Ahmad, & Wajid, 2019). In our analysis, we extracted 40 MFCC for each heart-sound file, corresponding to the three listening points, by using a hop length of 512 with an FFT window length of 2048 samples. Table 2 reports a snapshot of the dataset structure, where each row corresponds to a 2-s audio segment containing 40 MFCC values. Thereby, 30 items/segments represent a 60-s heartbeat sample thus resulting in a 30 × 40 matrix considering the features. Our next step was to adapt the dataset structure in both time series and tabular data according to each specific ML model.

3. HVD machine learning classifier

This section focuses on the core component of the HVDD MCPS: the HVD Classifier. It is based on the process shown in Fig. 4, implementing the *Model Processing*, *Training*, *Testing* and *Validation* steps. Here the focus is on the *Model Processing*, while the other steps are described in the next section.

Table 1
Dataset partitioning.

Label	(DS_{Label}) *Dataset Cardinality* c_{Label}			*Classification Tasks*			
	Train&Test #	*Valid. #*	*Overall #*	*Status*	*Valve*	*Cause*	*Degree*
	c^T_{Label}	c^V_{Label}	c^O_{Label}	*H(0)/ D(1)*	*A(0)/ M(1)*	*St(0)/ Re(1)*	*Mo(0)/ Se(1)*
Total of subjects (*All*)	110	22	132	*	*	*	*
Healthy (*H*)	30	8	38	0	*	*	*
Diseased (*D*)	80	14	94	1	*	*	*
Diseased Aortic (*DA*)	41+6	7	54	1	0	*	*
Diseased Mitral (*DM*)	33	7	40	1	1	*	*
Diseased Moderate (*DMo*)	4	3	7	1	*	*	0
Diseased Severe (*DSe*)	66	11	77	1	*	*	1
Diseased Stenosis (*DSt*)	26+2	5	33	1	*	0	*
Diseased Regurgitation (*DRe*)	44+2	9	55	1	*	1	*
Diseased Aortic Moderate (*DAMo*)	3	2	5	1	0	*	0
Diseased Aortic Severe (*DASe*)	38	5	43	1	0	*	1
Diseased Aortic Stenosis (*DASt*)	25	4	29	1	0	0	*
Diseased Aortic Regurgitation (*DARe*)	16	3	19	1	0	1	*
Diseased Mitral Moderate (*DMMo*)	1	1	2	1	1	*	0
Diseased Mitral Severe (*DMSe*)	28	6	34	1	1	*	1
Diseased Mitral Stenosis (*DMSt*)	1+2	1	4	1	1	0	*
Diseased Mitral Regurgitation (*DMRe*)	28+2	6	36	1	1	1	*
Diseased Aortic Stenosis Moderate (*DAStMo*)	1	1	2	1	0	0	0
Diseased Aortic Stenosis Severe (*DAStSe*)	24	3	27	1	0	0	1
Diseased Aortic Regurgitation Moderate (*DAReMo*)	2	1	3	1	0	1	0
Diseased Aortic Regurgitation Severe (*DAReSe*)	14	2	16	1	0	1	1
Diseased Mitral Stenosis Moderate (*DMStMo*)	0	0	0	1	1	0	0
Diseased Mitral Stenosis Severe (*DMStSe*)	1	1	2	1	1	0	1
Diseased Mitral Regurgitation Moderate (*DMReMo*)	1	1	2	1	1	1	0
Diseased Mitral Regurgitation Severe (*DMReSe*)	27	5	32	1	1	1	1

Table 2
Sample data rows for each audio segment of 2 s.

Segment	MFCC_1	MFCC_2	…	MFCC_40
1	value1.1	value1.2	…	value1.40
2	value2.1	value2.2	…	value2.40
⋮	⋮	⋮	⋮	⋮
30	value30.1	value30.2	…	value30.40

3.1. Model structures

The HVDD problem is a multi-class classification problem. Based on the dataset description of Section 2.2.1, 9 classes (1 normal, 8 abnormal) are identified, a large number to be inferred by a dataset of just 132 entries. To deal with such an issue, the main classification problem has been split into simpler *classification subproblems* or *tasks* in a top-down/divide and conquer strategy, adopting combined models and ensemble learning techniques. More specifically,

Definition 3.1. a *classification problem* cp is a pair

$$cp = \{\mathbf{DS}, \mathbf{CL}\}$$

where:

- **DS** is the *input dataset* with all data items to be classified;
- $\mathbf{CL} = \{cl_1, \dots, cl_n\}$ is the *set of classes* associated with the classification task, with $n \geq 2$ ($n = 2$ for binary classification tasks).

A classification problem cp can be decomposed into $m > 1$ *classification subproblems or tasks* ct_i, i.e.

$$cp = \{ct_1, \dots, ct_m\}$$

where $ct_i = \{\mathbf{DS}_i, \mathbf{CL}_i\} \in cp$, such that $\mathbf{DS}_i \subseteq \mathbf{DS}$, and $\mathbf{CL}_i \subseteq \mathbf{CL}$, $\bigcup_i \mathbf{CL}_i = \mathbf{CL}$, $\mathbf{CL}_i \bigcap \mathbf{CL}_j = \emptyset \ \forall i, j = 1, \dots, m;\ i \neq j$, thus resulting in the classification problem set of classes partitioning. Thereby, it can be argued that there are several ways to decompose a given classification problem into classification tasks, i.e. the *decomposition* step. From the set theory, an n-class classification problem cp can be decomposed into $B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k$ total number of partitions, i.e. the Bell number, which grows combinatorially with n. This number also considers sets with only 1 element not satisfying Definition 3.1 since each classification task ct_i must be at least binary ($n_i \geq 2$), and the relationships between classes can be taken into account in the partitioning, e.g. the status (healthy-diseased), valve (aortic-mitral), cause (stenosis-regurgitation), and degree (moderate-severe). However, from the above, it is possible to qualitatively argue that such a number grows combinatorially.

The $ct_i = \{\mathbf{DS}_i, \{cl_1, \dots, cl_n\}\}$ classification task (as well as the $cp = \{\mathbf{DS}, \mathbf{CL}\}$ problem) solution can be represented as an *inference function* $f(ct_i)$ ($f(cp)$ for cp) applied to the task (or the problem) resulting in the input dataset $\mathbf{DS}_i$ (**DS**) partition into class (sub)datasets as follows:

$$f : \{\mathbf{DS}_i, \{cl_1, \dots, cl_n\}\} \rightarrow \{\mathbf{DS}_{i,1}, \dots, \mathbf{DS}_{i,n}\}$$

where, for the set partitioning rules, $\mathbf{DS}_{i,j} \neq \emptyset \subseteq \mathbf{DS}_i$ is a subset of the input dataset $\mathbf{DS}_i$ only containing data items of class j, and $\bigcup_i \mathbf{DS}_{i,j} = \mathbf{DS}_i$, $\mathbf{DS}_{i,h} \bigcap \mathbf{DS}_{i,k} = \emptyset \ \forall h, k = 1, \dots, n;\ h \neq k$.

The classification problem here considered is the 9-class (H, DAStMo, DAStSe, DAReMo, DAReSe, DMStMo, DMStSe, DMReMo, and DMReSe) problem cp_{HVD} on the above described dataset (DS_{All}). This can be solved by adopting several different partitioning as discussed above, which results into *model structures* properly combining them, mainly hierarchically into *levels*.

Thereby, the *model processing* phase consists of three main steps: (i) the *model structure selection*, (ii) the *model structure decomposition* and (iii) the *dataset partitioning*. The *model structure selection* concerns the choice of the model structure among the five alternatives (M1–M5) specified in Fig. 5. This process, mainly driven by the dataset size, aims to select the best model structure (i.e. the one with highest accuracy) for the heart valve disease diagnosis (some selection guidelines are discussed in Section 6). Once the model structure is selected, it is necessary to split it into the classification tasks. The *model structure decomposition* is therefore tasked at this, i.e. identifying all the classification tasks of the selected model structure, corresponding to a tree of models, which should be processed by a level order traversal or breadth first search (BFS), where higher level task have to be processed before lower level ones, and same level tasks can be even processed

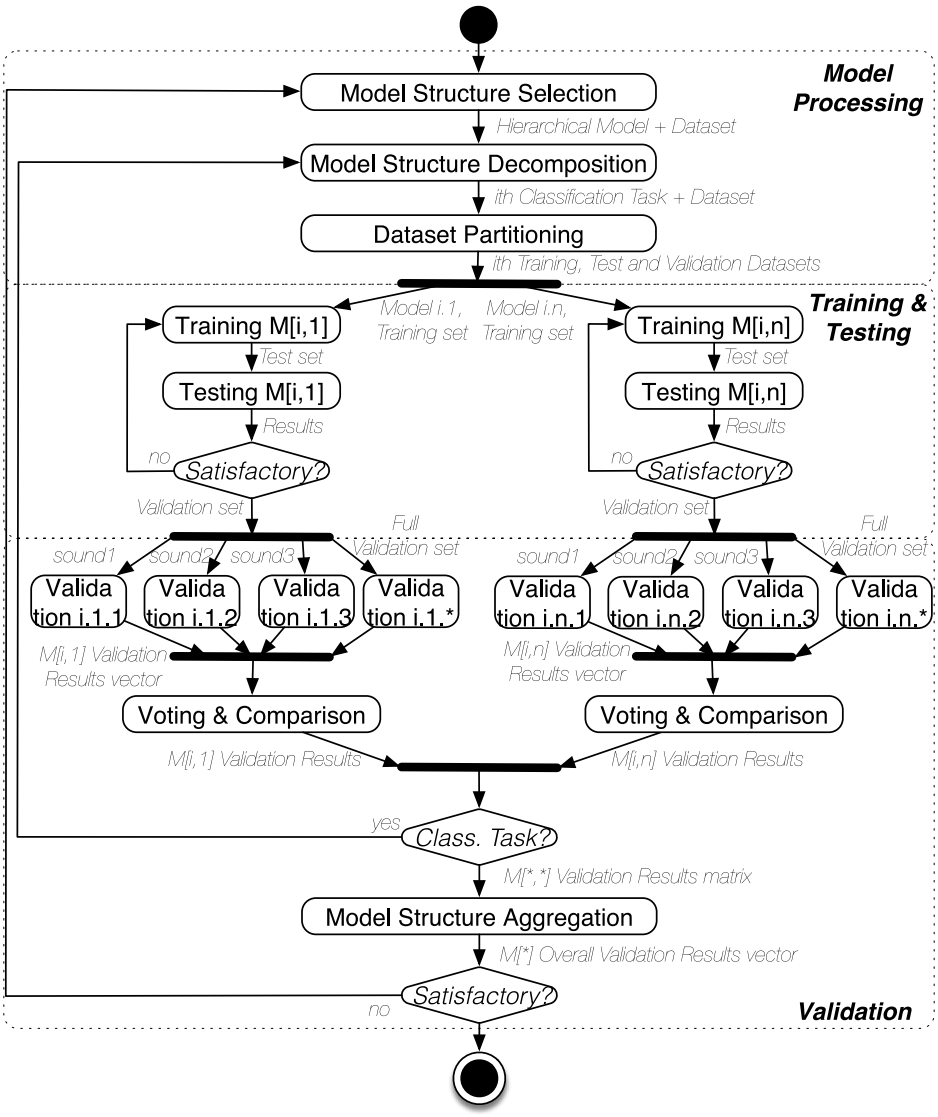


Fig. 4. The HVDD training workflow.

simultaneously in parallel. Once the model structure is decomposed, the *dataset partitioning* is performed, to set up the dataset required by each of the classification tasks composing of the selected model structure. All such model structures start from the full dataset (DS_{All}), then refined into specific sub-dataset (DS_D, DS_{DA}, DS_{DM}, etc.) in accordance with the classification task to be performed. In Fig. 5, the grey datasets characterize the lack of data to perform the corresponding classification task (i.e. DS_{DMStMo} in the model structures 1 and 2).

In Fig. 5, five *model structures* are identified to solve the cp_{HVD} problem, thus resulting in trees of (binary or multiclass) classification tasks to be then combined and solved following a level order traversal. Besides several other relevant partitioning and model structures can be defined, as also discussed above, the rationale behind this 5 model choice is based on two main principles: (i) to have a feasible model structure based on the dataset available, (ii) to compare the model performance on different options and parameters (number of levels, classes, accuracy) to get some guidelines (reported in Section 6.2).

In this light, Fig. 5 model structure 1 M_1 implements a flat (single level) structure based on the 9-class above listed, the single level/task problem that is the starting point of any further decomposition. To overcome the lack of data issue, this classification problem cp_{HVD} is indeed decomposed into classification tasks, identifying multiple-level *hierarchical* models. The *model structure 2* M_2 (top left of Fig. 5) is obtained by recursively adopting binary classification tasks, thus resulting into 4 levels: $(DS_{All}, (cl_H, cl_D))$ at level 1, $(DS_D, (cl_{DA}, cl_{DM}))$ at level 2, $(DS_{DA}, (cl_{DASt}, cl_{DARe}))$ and $(DS_{DM}, (cl_{DMSt}, cl_{DMRe}))$ at level 3, and four other tasks at level 4 $(DS_{DASt}, (cl_{DAStMo}, cl_{DAStSe}))$, $(DS_{DARe}, (cl_{DAReMo}, cl_{DAReSe}))$, $(DS_{DMSt}, (cl_{DMStMo}, cl_{DMStSe}))$, and $(DS_{DMRe}, (cl_{DMReMo}, cl_{DMReSe}))$. Unfortunately, the current dataset has not enough data to enact M_2 (mainly for the level 4 classification tasks $(DS_{DMSt}, (cl_{DMStMo}, cl_{DMStSe}))$).

A feasible solution is to further aggregate classes and classification tasks as done, for example, in the model structures 3, 4, and 5. The *model structure 3* M_3 (bottom left of Fig. 5), also adopts only binary classification tasks, combining them with a different logic into three levels: $(DS_{All}, (cl_H, cl_D))$ for level 1, $(DS_D, (cl_{DA}, cl_{DM}))$ for level 2, and $(DS_{DA}, (cl_{DAMo}, cl_{DASe}))$, $(DS_{DA}, (cl_{DASt}, cl_{DARe}))$ for the aortic disease, and $(DS_{DM}, (cl_{DMMo}, cl_{DMSe}))$ $(DS_{DM}, (cl_{DMSt}, cl_{DMRe}))$ for the mitral disease in level 3.

The *hierarchical model structure 4* M_4 (top right of Fig. 5), has been designed by reducing the number of layers (to 2) and introducing multiclass tasks. Thereby a 3-class classification task is implemented at level 1, discriminating among healthy, aortic or mitral diseased subjects $(DS_{All}, (cl_H, cl_{DA}, cl_{DM}))$, while at level 2, the same classification tasks of the model structure 3 M_3-level 3 are exploited.

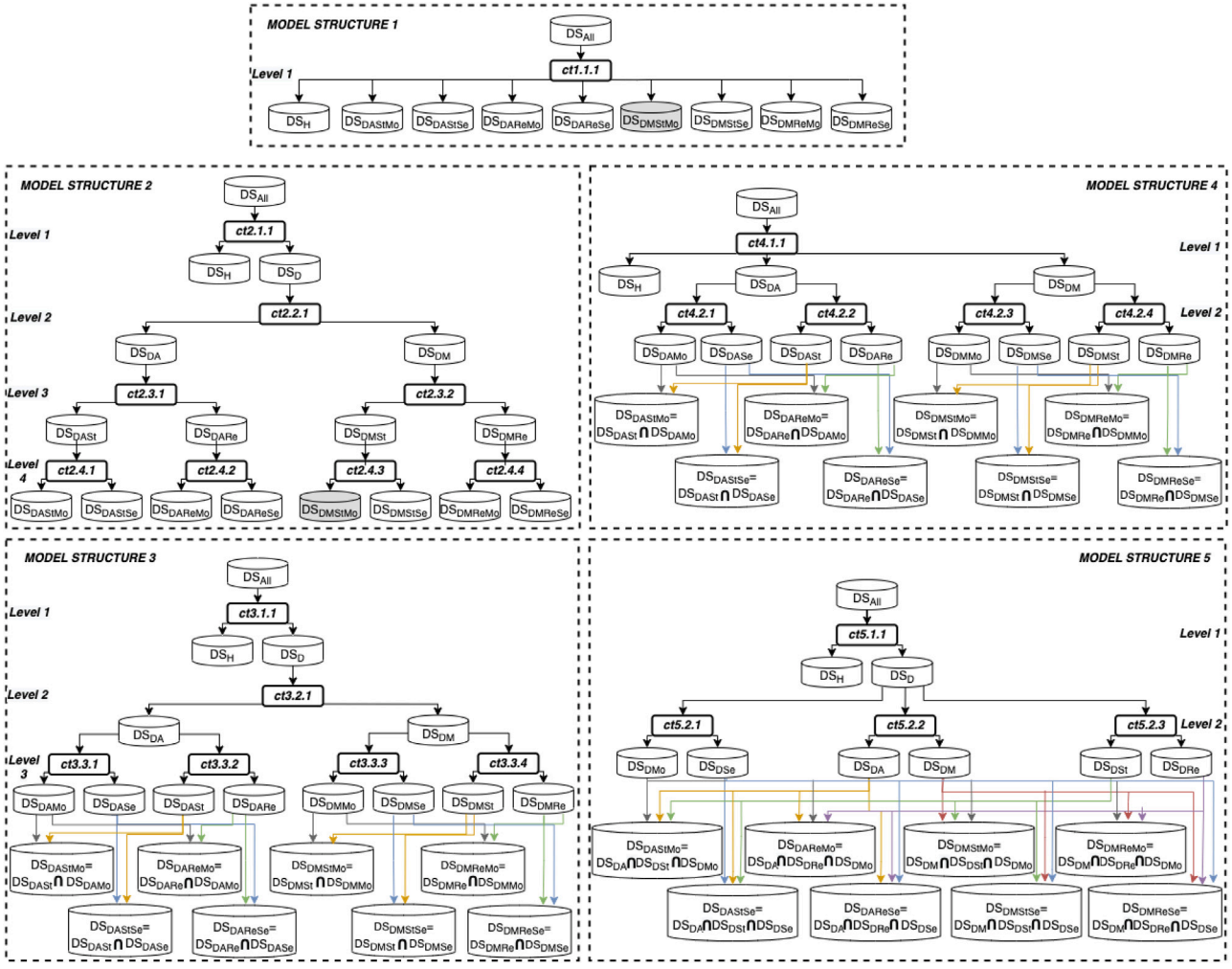


Fig. 5. HVDD model structures, classification levels and tasks.

The *hierarchical model structure 5* M_5 (bottom right of Fig. 5) is selected to assess and compare the impact of the class parameter (binary vs. multiclass) in the model structure, thus compared to M_4, as well as the impact of levels (M_4 and M_5 vs. M_3). To such a purpose, M_5 is only composed of (4) binary classification tasks organized in 2 levels: $(DS_{All}, (cl_H, cl_D))$ at level 1 and $(DS_D, (cl_{DMo}, cl_{DSe}))$, $(DS_D, (cl_{DA}, cl_{DM}))$, and $(DS_D, (cl_{DSt}, cl_{DRe}))$ at level 2.

M_3, M_4, and M_5, however need further processing to obtain the results, which is performed by the aggregation step combining such results as shown in the bottom of the corresponding model structures in Fig. 5 as detailed in Section 4.3.

3.2. Model types

Thus, each HVD classification task has to be solved by one or more ML techniques. Different supervised ML techniques have been considered in this paper, in order to assess and compare different solutions, here identified as *model types*. ML and DL model types have been demonstrated to be quite effective in monitoring heartbeat sounds, able to recognize murmurs characterizing heart valve diseases. As mentioned in Section 2.2.2, we have extracted and used MFCCs features of heartbeat audio signals by building and training different models for distinct levels of classification tasks. Specifically, we selected, tested and compared six ML classifiers or model types: Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes (NB), Logistic Regression (LR), multilayer feedforward deep neural networks (DNN) and Long-Short Term Memory (LSTM), a Recurrent Neural Network (RNN). RF is a model type widely used in classification problems, with specific reference to heart disease diagnostic (see, for example Masetic and Subasi (2016)). It is based on a set of decision trees working on randomly selected subsets of the training dataset, then aggregating their classification results, thus identifying the proper class by a voting mechanism (Pal, 2005). SVM is another supervised learning model commonly used in linear/nonlinear classification disease problems (Austin, Tu, Ho, Levy, & Lee, 2013), based on the construction of a class separator line on a hyperplane aiming to maximize the margin between points on both sides of the decision line (Gokulnath & Shantharajah, 2019). NB is also frequently used in medical analysis (Pattekari & Parveen, 2012), starting from the Bayes theorem for classifying data points, using conditional probability (Subbalakshmi, Ramesh, & Rao, 2011) for the classification. LR is a classical ML algorithm widely used in the medical applications for binary and linear classification problems (Khemphila & Boonjing, 2010), and it is based on a logit function to generate the probability of a discrete outcome given an input variable (Edgar & Manz, 2017).

A classical multilayer feedforward DNN model (input nodes, two or more hidden layers and the output nodes) suitable for classification problems in medicine (Darmawahyuni, Nurmaini, & Firdaus, 2019) has been also investigated. DNN models are able to deal with data *nonlinearity*, but considering that there are no feedback connections (outputs of the model are not fed back into it), usually, they suffer from overfitting problems (Cogswell, Ahmed, Girshick, Zitnick, & Batra, 2015).

To deal with overfitting, RNN are a possible solution by using or even combining *regularization*, *dropout*, *early stopping*, *data augmentation*, and/or *batch normalization* techniques (Tian & Zhang, 2022). In addition, RNN can store/remember previous inputs in their memory (Pascanu, Gulcehre, Cho, & Bengio, 2013). Specifically, we used LSTM networks (Hochreiter & Schmidhuber, 1997), keeping in memory temporal dependencies for long sequences and time frames, to allow recognizing key data patterns more effectively than other deep neural networks. For such a reason, the LSTM model type is used in many time-dependent applications such as speech, image, and video processing (Smagulova & James, 2019).

The rationale behind an LSTM network is to add layers in cascade to improve feature extraction and classification (Bruneo & De Vita, 2019; Gokmen, Rasch, & Haensch, 2018). The core of such an LSTM is the *cell state*, basically able to add or remove information from the three *input*, *forget* and *output gates*, tasked at reading the current input, forgetting the current cell state value, providing the output of the current cell value, respectively. Thereby, it takes as input the data points falling into the corresponding time window together with the cell state and the output of the previous step. Then, a *dense layer* generates the label representing the heart valve disease class inferred, potentially outperforming classical ML approaches through the double (long-/short-term) memory mechanism.

3.3. Dataset partitioning

The full dataset composed of heart sounds of all subjects (38 healthy and 94 diseased patients) has been split into two sequential partitioning stages: (i) *learning partitioning*, (ii) *model partitioning*. The first partitioning stage, i.e. the learning one, aims at decomposing the original dataset into three sub-datasets: *training*, *test* and *validation* datasets. Then, such datasets are further split by the *model partitioning* stage, considering the model structure and its classification tasks, as reported in Table 1. More specifically, although the learning partitioning is performed before the model partitioning, it should take into account the validation requirements at any of the levels of the hierarchical model, taking off the same dataset subjects from all the classification tasks to enable the overall model validation.

With specific regard to the learning partitioning, as reported in Table 1, 24 healthy and 64 diseased subjects have been selected for the training set, 6 healthy and 16 diseased subjects for the test set, and 8 healthy and 14 diseased subjects for the external validation dataset. According to the model structures of Fig. 5, each ML model type has been trained on a specific dataset, summarized in Table 1 by its label, type, classification task, train–test, validation and the overall number of subjects. Specifically, the classification tasks associated to a dataset can be represented as ordered *4-tuples* of binary elements $(0, 1)$

$$(H/D, A/M, S/R, Mo/Se) \in \mathbb{N}_2^4.$$

Each dataset can be associated with a set of classification tasks, represented in Table 1 by the * meaning that the corresponding tuple element can be either 0 or 1, thus identifying 2 classification tasks.

In the same Table 1, we have included subjects marked with the symbol "+" in the "Train&Test" column (e.g., +6, +2, etc.). These specific subjects have been intentionally chosen or excluded from the dataset due to their allocation to more than one distinct category. This approach was implemented to mitigate any potential training inaccuracies within each hierarchical classification model. For instance, the "+6" value (displayed in the "Train&Test" column for DA) pertains to six subjects labelled with aortic (A) conditions, involving both stenosis and regurgitation issues as well as moderate and severe degree. Due to the simultaneous presence of these conditions, these "+6" subjects have been utilized for D and DA while being omitted from the lower-level aortic disease datasets (DAMo, DASe, DASt, DARe, DAStMo, DAStSe, DAReMo, and DAReSe). The same principle is applied to the four subjects marked as "+2" in "Diseased Stenosis – DSt" and "+2" in "Diseased Regurgitation - DRe". These subjects are also encompassed in "Diseased Mitral Stenosis - DMSt" and "Diseased Mitral Regurgitation – DMRe", exhibiting both moderate and severe mitral issues. Consequently, they are not included in the corresponding subcategories (DMStMo, DMStSe, DMReMo, and DMReSe).

4. Training, testing, validation and aggregation

According to the workflow depicted in Fig. 4, after model processing steps, training, testing, validation and aggregation ones are performed to obtain a fully operating HVDD system as detailed in the following.

4.1. Training & Testing

The training and testing phases have been carried out once the model structure has been selected and then decomposed into classification tasks. Each of such classification tasks is thus performed by exploiting different ML models to compare their results and effectiveness. The classification task dataset then identified by the model partitioning stage, is consequently split based on the learning process with 80% of the data as training set and the remaining 20% as the test set.

Table 3 shows the summary of the hyperparameters (with ranges and optimal values) of all the ML models exploited in the training. The consequent testing phase is based on a five k-fold cross-validation scheme (Bengio & Grandvalet, 2003). The ML model learning performance after training and testing has been thus assessed by the *Accuracy (Acc ± std)* (%), *Sensitivity (Sens)*, *Specificity (Spec)*, *Precision (Prec)*, *Recall (Rec)* and *F1-score (F1)* classification criteria (Uçar, Nour, Sindi, & Polat, 2020).

4.2. Validation

The trained models then obtained are validated by a specific validation dataset composed of subjects and data items not involved in the training and test stages, as discussed in Section 3.3. More specifically, the three heart sounds (aortic — sound 1, mitral — sound 2, and tricuspid — sound 3) of the validation dataset have been considered separately in the assessment of the accuracy metrics of the ML models (trained on the three heart sounds, altogether). Thereby, to improve the classification accuracy, a majority *voting* approach has been adopted on the three classification results corresponding to the three heart sounds of a subject in the validation dataset.

For each classification task, the result is then obtained by a voting mechanism applied to the three sound classification results for a subject. In particular, in the H/D binary classification of model structure 3 and model structure 5, a 3/3 voting mechanism has been adopted to classify *healthy* subjects, while a 1 out of 3 (1/3) mechanism for *diseased* subjects, respectively. In the model structure 4, the multiclass (H/DA/DM) classification task adopted a 2/3 majority voting policy, while in all other binary classification tasks, a 2/3 majority voting mechanism is applied to the three sound data items of each subject. The results then obtained are also compared against the one obtained without distinguishing on the three sounds, mixing their samples for each subject.

Once the voting mechanism has been performed for all classification tasks according to the specific model structure, the model structure aggregation step is performed. It consists of the aggregation of classification task results based on the HVDD model structure, to provide insights for the global decision-making process. Table 4 defines the accuracy formulae for the HVDD model structures above discussed (M1, M2, M3, M4, M5), where $w_{l_i} \in [0,1] \subset \mathbb{R}$ is the *weight* of the classification task ith class with l_i label ($i = 1,..,n_x$, n_x the number of classes of the classification tasks). w_{l_i} weights the ith class in the n_x-class classification task and can thus be considered as a probability to have on outcome of that class based on the previous outcomes (if known), and therefore $\sum_{i=1}^{n_x} w_{l_i} = 1$. Two types of weight are considered:

Table 3
Optimal hyperparameters of ML models.

Models	Hyperparameters	Range	Optimal parameter
LSTM	*Windows size*	[1, 30]	10
	N° of layers	[1, 5]	3
	N° unit for layer	[64,1028]	[512, 256, 128]
	Learning rate	[0.0001, 0.01]	0.001
	Training epochs	[20, 500]	200
	Dropout	[0.005, 0.3]	0.05
	Recurrentdropout	[0.005, 0.3]	0.20
	Optimizer	[*rmsprop*, *adam*]	*rmsprop*
	Early Stopping patience	[2, 10]	5
	Batch size	[16, 256]	32
	Hidden Activation function	[Relu, leakyrelu]	Relu
DNN	*N° of layers*	[1, 5]	3
	N° unit for layer	[64, 1028]	[512, 256, 128]
	Learning rate	[0.0001, 0.01]	0.001
	Training epochs	[20, 500]	200
	Dropout	[0.005, 0.3]	0.05
	Optimizer	[*rmsprop*, *adam*]	*rmsprop*
	Early Stopping patience	[2, 10]	5
	Batch size	[16, 256]	32
	Weight decay	[L1, L2]	L2
	Hidden Activation function	[Relu, leakyrelu]	Relu
	Output Activation function	[Sigmoid, Tahn]	Sigmoid (for binary)
	Output Activation function	[Softmax, Log Softmax]	Softmax (for multiclass)
SVM	*C*	[1, 10]	1.0
	kernel	['linear', 'poly', 'rbf', 'sigmoid']	'linear'
	gamma	['scale', 'auto']	'scale'
	coef	[0.0, 0.1]	0.0
	shrinking	[False, True]	True
RF	*n_estimators*	[100, 500]	100
	max_features	['auto', 'sqrt', 'log2']	'auto'
	max_depth	[10, 50]	'auto'
	criterion	['gini', 'entropy']	'entropy'
	min_samples_split	[2, 20]	2
	min_samples_leaf	[1, 15]	2
NB	*n_neighbors*	[1, 10]	5
	weights	['uniform', 'distance']	'uniform'
	algorithm	['auto', 'ball_tree', 'kd_tree', 'brute']	'auto'
	leaf_size	[10, 50]	30
LR	*penalty*	['l1', 'l2', 'elasticnet', 'none']	'l2'
	C	[1, 10]	1.0
	tol	[1e−8, 1e−3]	1e−4

Table 4
HVDD model structure accuracy formulae.

M1	$\sum_{i=1}^{n_x} w_{l_i} * a_{l_i}$
M2	$w_H a_H + w_D a_D \left(w_{DA} a_{DA} \left(w_{DASt} a_{DASt} (w_{DAStMo} a_{DAStMo} + w_{DAStSe} a_{DAStSe}) + w_{DARe} a_{DARe} (w_{DAReMo} a_{DAReMo} + w_{DAReSe} a_{DAReSe}) \right) + w_{DM} a_{DM} \left(w_{DMSt} a_{DMSt} (w_{DMStMo} a_{DMStMo} + w_{DMStSe} a_{DMStSe}) + w_{DMRe} a_{DMRe} (w_{DMReMo} a_{DMReMo} + w_{DMReSe} a_{DMReSe}) \right) \right)$
M3	$w_H a_H + w_D a_D \left(w_{DA} a_{DA} (w_{DAMo} a_{DAMo} + w_{DASe} a_{DASe}) (w_{DASt} a_{DASt} + w_{DARe} a_{DARe}) + w_{DM} a_{DM} (w_{DMMo} a_{DMMo} + w_{DMSe} a_{DMSe}) (w_{DMSt} a_{DMSt} + w_{DMRe} a_{DMRe}) \right)$
M4	$w_H^{(3)} a_H + w_{DA}^{(3)} a_{DA} (w_{DAMo} a_{DAMo} + w_{DASe} a_{DASe}) (w_{DASt} a_{DASt} + w_{DARe} a_{DARe}) + w_{DM}^{(3)} a_{DM} (w_{DMMo} a_{DMMo} + w_{DMSe} a_{DMSe}) (w_{DMSt} a_{DMSt} + w_{DMRe} a_{DMRe})$
M5	$w_H a_H + w_D a_D (w_{DA} a_{DA} + w_{DM} a_{DM}) \left(w_{DSt} a_{DSt} (w_{DMo} a_{DMo} + w_{DSe} a_{DSe}) + w_{DRe} a_{DRe} (w_{DMo} a_{DMo} + w_{DSe} a_{DSe}) \right)$

1. *unbiased* if the classification task is performed without any preliminary knowledge, *a-priori*, so that

$$w_{l_i} = w_{l_j} = 1/n_x$$

$\forall i, j = 1, .., n_x$, (e.g. $w_{l_1} = w_{l_2} = 0.5$ for binary classification tasks),

2. *biased* if the classification task has a preliminary knowledge of the dataset, knowing *a-posteriori* the cardinality of the output dataset from the previous classification outcomes. Specifically,

$$w_{l_i} = \frac{c_{l_i}}{\sum_{j=1}^{n_x} c_{l_j}} = \frac{c_{l_i}}{c_x}$$

where c_{l_j} is the cardinality of the l_j output dataset (with $j = 1, \ldots, n_x$, and $\sum_{j=1}^{n_x} c_{l_j} = c_x$).

Table 5 shows the values of the two types of weight (biased and unbiased) computed with the formulas described above. Then, a_{l_i} is the accuracy of each class in the classification task.

4.3. Model structure aggregation

Once the models are trained, tested and validated, the focus is shifted to the model structure selected in the beginning of Fig. 4 workflow. According to the latter, the final step to be performed is the aggregation one, thus stepping back to the model structure to get

Table 5
Biased weight values for the accuracy formulae of Table 4.

w_{l_i}	w_H	w_D	$w_H^{(3)}$	$w_{DA}^{(3)}$	$w_{DM}^{(3)}$	w_{DA}	w_{DM}	w_{DSt}	w_{DRe}	w_{DMo}
B	0.36	0.64	0.36	0.32	0.32	0.50	0.50	0.36	0.64	0.21
UB	0.50	0.50	0.33	0.33	0.33	0.50	0.50	0.50	0.50	0.50
w_{l_i}	w_{DSe}	w_{DAMo}	w_{DASe}	w_{DASt}	w_{DARe}	w_{DMMo}	w_{DMSe}	w_{DMSt}	w_{DMRe}	
B	0.79	0.29	0.71	0.57	0.43	0.14	0.86	0.14	0.86	
UB	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	

the final results. Specifically, in the training stage, the aggregation step returns the n ($n = 9$ in the case study of Fig. 5) datasets categorizing and splitting the initial dataset (DS_{All}) items into n (sub-)datasets (DS_H, DS_{DAStMo}, DS_{DAStSe}, DS_{DAReMo}, DS_{DAReSe}, DS_{DMStMo}, DS_{DMStSe}, DS_{DMReMo}, DS_{DMReSe}). In the inference stage, the aggregation step only returns the specific data item class.

This process, however, strongly depends on the model structure in the case a partitioning scheme is adopted and the overall results have to be obtained by combining the lower level classification task results (e.g. for M_3, M_4 and M_5 in the case study of Fig. 5). In such cases, indeed, the specific aggregation rules have to be defined to get the final results, based on the initial decomposition. Thus, this process, coupling with the partitioning one, is hard to be automated and thus manually performed by the designer/partitioner.

In the HVDD case study of Fig. 5, the three model structure returns 9 datasets after the training–test–validation which are combined as shown in Table 6. The inference process is thus performed accordingly: if for example a patient affected by moderated aortic stenosis ($DAStMo$) is processed, the M_3 classification tasks at level 3 are activated simultaneously and returns $DAMo$ ($ct3.3.1$) and $DASt$ ($ct3.3.2$). Similarly, M_4 $ct4.2.1$ returns $DAMo$ while $ct4.2.2$ returns $DASt$. In M_5, 3 classification tasks are simultaneously activated: $ct5.2.1$ returns DMo, $ct5.2.2$ returns DA, and $ct5.2.3$ returns DSt.

5. Experimental results

This section reports on the ML metric assessment of the classifiers above discusses, both in test and validation. For the experimental part, we choose Python as runtime environment and Scikit-learn (Pedregosa et al., 2011) and Keras libraries (Chollet et al., 2018) for the implementation and training of all ML classifiers. Audio processing and all experiments were performed on GPU-based machines equipped with 3 GeForce GTX TITAN X (12 GB) GPUs, provided by the Messina IRIB-CNR partner of this study.

5.1. Test results

As mentioned in Section 4.1, the ML model metrics *accuracy*, *precision*, *recall* and *F1-score* Table 7 shows the ML/DL model performance metrics (i.e. *accuracy*, *precision*, *recall* and *F1-score*) obtained by the test after training, highlighting that the LSTM outperformed the other models in terms of accuracy. In particular, the LSTM model has a high degree of accuracy in binary classification tasks such as H/D (98.07%) and DA/DM (97.64%). Furthermore, the LSTM model has also shown a high degree of accuracy in multi-class classification tasks such as H/DA/DM (92.45%), more complex than binary classification tasks, usually requiring more data and more complex models to achieve good performance. Another important finding is that the LSTM model has high accuracy in classifying Stenosis/Regurgitation (St/Re) binary classification (95.73%) by using all diseased subjects, and into binary classification of Stenosis/Regurgitation both for Aortic (DASt/DARe) (91.48%) and Mitral (DMSt/DMRe) (89.99%). This is particularly relevant since an accurate diagnosis of stenosis and/or regurgitation is crucial for the treatment and management of heart diseases, usually not taken into account by the previous work on HVD diagnosis. The LSTM model performance are slightly lower than the RF one for both Aortic and Mitral Moderate/Severe classification tasks (DAMo/DASe - 94.77% vs. 95.00%) and (DMMo/DMSe - 94.27% vs. 97.76%).

In addition, an in-depth analysis to evaluate the performance of the ML models in classifying diseased and healthy diagnoses (binary H/D) has been performed exploiting the *receiver operating characteristic* (ROC) curve, which plots the sensitivity and 1-specificity of the testing set. The ROC curve enables the visualization of the trade-off between these two measures, and it is a common method to evaluate the performance of a binary classifier. The ROC curves plotted in Fig. 6 confirm that the LSTM model is the best among the considered ones, showing the best balance between sensitivity and specificity.

Table 6
Aggregation rules of the HVDD model structures M_3, M_4, and M_5.

	DS_{DAStMo}	DS_{DAStSe}	DS_{DAReMo}	DS_{DAReSe}	DS_{DMStMo}	DS_{DMStSe}	DS_{DMReMo}	DS_{DMReSe}
M_3	$DS_{DASt} \cap DS_{DAMo}$	$DS_{DASt} \cap DS_{DASe}$	$DS_{DARe} \cap DS_{DAMo}$	$DS_{DARe} \cap DS_{DASe}$	$DS_{DMSt} \cap DS_{DMMo}$	$DS_{DMSt} \cap DS_{DMSe}$	$DS_{DMRe} \cap DS_{DMMo}$	$DS_{DMRe} \cap DS_{DMSe}$
M_4	$DS_{DASt} \cap DS_{DAMo}$	$DS_{DASt} \cap DS_{DASe}$	$DS_{DARe} \cap DS_{DAMo}$	$DS_{DARe} \cap DS_{DASe}$	$DS_{DMSt} \cap DS_{DMMo}$	$DS_{DMSt} \cap DS_{DMSe}$	$DS_{DMRe} \cap DS_{DMMo}$	$DS_{DMRe} \cap DS_{DMSe}$
M_5	$DS_{DA} \cap DS_{DSt} \cap DS_{DMo}$	$DS_{DA} \cap DS_{DSt} \cap DS_{DSe}$	$DS_{DA} \cap DS_{DRe} \cap DS_{DMo}$	$DS_{DA} \cap DS_{DRe} \cap DS_{DSe}$	$DS_{DM} \cap DS_{DSt} \cap DS_{DMo}$	$DS_{DM} \cap DS_{DSt} \cap DS_{DSe}$	$DS_{DM} \cap DS_{DRe} \cap DS_{DMo}$	$DS_{DM} \cap DS_{DRe} \cap DS_{DSe}$

Table 7
Performance metrics of all ML models on the test set in terms of Accuracy (Acc ± std), Precision (Prec), Recall (Rec), F1-score (F1).

Class.	LSTM				DNN				RF				SVM				NB				LR			
tasks	*Acc*(±*std*)	*Prec*	*Rec*	*F1*	*Acc*(±*std*)	*Prec*	*Rec*	*F1*	*Acc*(±*std*)	*Prec*	*Rec*	*F1*	*Acc*(±*std*)	*Prec*	*Rec*	*F1*	*Acc*(±*std*)	*Prec*	*Rec*	*F1*	*Acc*(±*std*)	*Prec*	*Rec*	*F1*
(DS_{All}, (cl_H, cl_D))	**0.9807 (±0.008)**	0.99	0.99	0.99	0.9324 (±0.012)	0.94	0.94	0.94	0.9652 (±0.005)	0.96	0.96	0.96	0.9333 (±0.007)	0.93	0.93	0.93	0.8337 (±0.009)	0.85	0.85	0.85	0.9238 (±0.004)	0.93	0.93	0.93
(DS_D, (cl_A, cl_M))	**0.9764 (±0.006)**	0.99	0.99	0.99	0.7080 (±0.056)	0.79	0.79	0.79	0.8391 (±0.016)	0.86	0.86	0.86	0.5993 (±0.022)	0.60	0.60	0.60	0.6108 (±0.022)	0.62	0.62	0.62	0.5974 (±0.012)	0.61	0.61	0.61
(DS_{All}, (cl_H, cl_A, cl_M))	**0.9245 (±0.023)**	0.95	0.95	0.95	0.7408 (±0.016)	0.75	0.75	0.75	0.8171 (±0.013)	0.84	0.84	0.83	0.5351 (±0.014)	0.53	0.53	0.53	0.5264 (±0.010)	0.52	0.52	0.51	0.5233 (±0.009)	0.51	0.51	0.51
(DS_D, (cl_{St}, cl_{Re}))	**0.9573 (±0.019)**	0.97	0.97	0.97	0.7553 (±0.013)	0.80	0.76	0.76	0.8803 (±0.010)	0.91	0.91	0.91	0.6899 (±0.014)	0.67	0.68	0.67	0.6380 (±0.016)	0.62	0.63	0.62	0.6813 (±0.009)	0.69	0.69	0.69
(DS_{DA}, (cl_{St}, cl_{Re}))	**0.9148 (±0.010)**	0.95	0.95	0.95	0.8778 (±0.012)	0.89	0.89	0.89	0.8945 (±0.004)	0.93	0.93	0.93	0.7937 (±0.006)	0.79	0.79	0.79	0.6430 (±0.020)	0.64	0.64	0.64	0.7451 (±0.022)	0.79	0.78	0.78
(DS_{DM}, (cl_{St}, cl_{Re}))	**0.8999 (±0.008)**	0.93	0.92	0.92	0.8333 (±0.001)	0.84	0.83	0.83	0.8825 (±0.012)	0.90	0.91	0.91	0.8263 (±0.039)	0.85	0.85	0.84	0.7638 (±0.021)	0.73	0.74	0.74	0.8388 (±0.027)	0.77	0.78	0.77
(DS_D, (cl_{Mo}, cl_{Se}))	**0.9516 (±0.031)**	0.97	0.97	0.97	0.9128 (±0.032)	0.93	0.93	0.93	0.9470 (±0.008)	0.94	0.94	0.94	0.8045 (±0.012)	0.79	0.79	0.74	0.7186 (±0.029)	0.75	0.74	0.74	0.7957 (±0.008)	0.79	0.81	0.79
(DS_{DA}, (cl_{Mo}, cl_{Se}))	0.9477 (±0.029)	0.98	0.98	0.98	0.9240 (±0.035)	0.97	0.96	0.97	**0.9500 (±0.011)**	0.96	0.96	0.96	0.8250 (±0.013)	0.85	0.85	0.82	0.7344 (±0.009)	0.77	0.76	0.77	0.8372 (±0.005)	0.80	0.82	0.78
(DS_{DM}, (cl_{Mo}, cl_{Se}))	0.9427 (±0.012)	0.97	0.95	0.96	0.9368 (±0.037)	0.97	0.97	0.97	**0.9776 (±0.004)**	0.99	0.99	0.99	0.9497 (±0.003)	0.90	0.92	0.90	0.7927 (±0.025)	0.94	0.80	0.85	0.9236 (±0.003)	0.90	0.92	0.90

Table 8
Validation single (Si) vs. majority-voting (Vot) accuracy results.

Classification	LSTM	DNN	RF	SVM	NB	LR
Task	Si | Vot	Si | Vot	Si | Vot	Si | Vot	Si | Vot	Si | Vot
$(DS_{All}, (cl_H, cl_D))$	0.84 | 1.00	0.73 | 0.81	0.78 | 0.91	0.77 | 0.91	0.72 | 0.82	0.80 | 0.86
$(DS_D, (cl_A, cl_M))$	0.69 | 1.00	0.62 | 0.79	0.59 | 0.78	0.52 | 0.50	0.49 | 0.42	0.55 | 0.57
$(DS_{All}, (cl_H, cl_A, cl_M))$	0.68 | 0.91	0.55 | 0.73	0.62 | 0.73	0.62 | 0.68	0.55 | 0.59	0.60 | 0.59
$(DS_D, (cl_{St}, cl_{Re}))$	0.69 | 0.86	0.78 | 0.78	0.63 | 0.78	0.65 | 0.78	0.64 | 0.71	0.65 | 0.64
$(DS_{DA}, (cl_{St}, cl_{Re}))$	0.73 | 0.71	0.61 | 0.57	0.69 | 1.00	0.74 | 0.86	0.76 | 0.85	0.74 | 0.85
$(DS_{DM}, (cl_{St}, cl_{Re}))$	0.85 | 1.00	0.90 | 1.00	0.72 | 0.86	0.69 | 0.85	0.59 | 0.57	0.61 | 0.43
$(DS_D, (cl_{Mo}, cl_{Se}))$	0.75 | 1.00	0.79 | 0.93	0.87 | 1.00	0.94 | 1.00	0.65 | 0.79	0.97 | 1.00
$(DS_{DA}, (cl_{Mo}, cl_{Se}))$	0.72 | 0.71	0.69 | 0.86	0.82 | 1.00	0.99 | 1.00	0.51 | 0.57	0.97 | 1.00
$(DS_{DM}, (cl_{Mo}, cl_{Se}))$	1.00 | 1.00	0.98 | 1.00	0.98.| 1.00	0.99 | 1.00	0.85 | 0.86	0.99 | 1.00

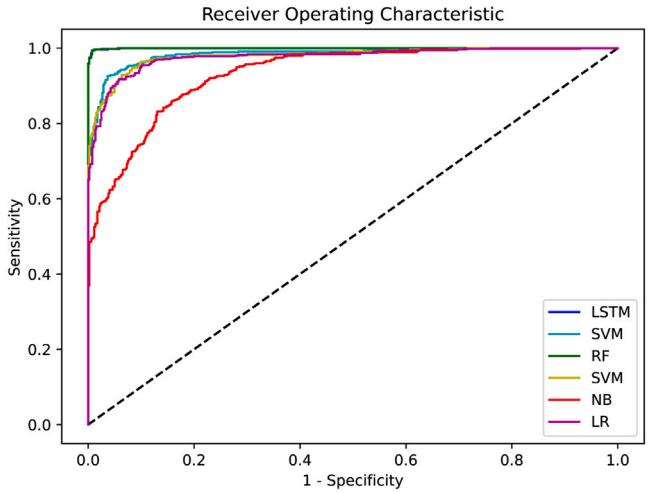


Fig. 6. ROC Curves on H/D binary classification.

5.2. Validation results

In the validation, the ML models are tested on an extra dataset, i.e. not included in the training and test sets, composed of 22 subjects. The results of the assessment, reported in Table 8, shows that the LSTM model outperforms the others in all classification tasks, while highlighting the majority voting effectiveness in increasing the accuracy. Voting techniques are applied, indeed, on the three different sounds, considering three different classification tasks then compared applying majority voting (Vot), while the single classification task is trained on a blended dataset mixing the three hearth source sounds, as explained in Section 4.2. Table 9 shows the ML model structures accuracy assessed by the formulae described in Section 4.2.

Also in this case, the LSTM model is the best performing, and specifically, the structure 5 shows the highest accuracy, i.e. 95.14% in the test set (T&T) and 93.00% in the validation set (Val) for the unbiased model, while the biased one is slightly lower (93.39% for T&T and 91.09% for Val).

5.3. Experiments on other public datasets

To further validate our MFCCs-based LSTM approach, we evaluated its performance on two well-known public datasets as baseline: (i) the Korean (Son & Kwon, 2018) and (ii) the PhysioNet/Computing in Cardiology (PhysioNet/CinC) (Liu et al., 2016) ones. The former involves subjects with a normal heart sound signal (H) and four multi abnormals diseases: Murmur in systole (MVP), Mitral Regurgitation (MR), Mitral Stenosis (MS), and Aortic Stenosis (AS), 1000 in total (200 for each class). On the other hand, the 2016 PhysioNet/CinC Challenge dataset includes 3240 heart sounds collected from healthy subjects (2575) and patients affected by different heart diseases including heart valve ones (665).

Table 9
Training & Test (T&T) as well as Validation (Val) accuracies of three model structures using Unbiased and Biased parameters.

Models	Model structure 3		Model structure 4		Model structure 5	
	T&T	Val	T&T	Val	T&T	Val
Unbiased						
LSTM	**0.9235**	**0.8760**	**0.8761**	**0.7596**	**0.9514**	**0.9300**
DNN	0.7664	0.6434	0.6649	0.6059	0.7225	0.6371
RF	0.8404	0.7851	0.7597	0.6959	0.8253	0.7318
SVM	0.6701	0.6495	0.4447	0.6143	0.6190	0.6325
NB	0.5713	0.4939	0.3729	0.3884	0.5478	0.5066
LR	0.6651	0.5869	0.4179	0.4484	0.6196	0.5869
Biased						
LSTM	**0.8956**	**0.8422**	**0.8716**	**0.7664**	**0.9339**	**0.9109**
DNN	0.6897	0.5980	0.6595	0.6116	0.6236	0.5899
RF	0.7851	0.7510	0.7524	0.6975	0.7641	0.6832
SVM	0.5475	0.5785	0.4316	0.6173	0.4776	0.5568
NB	0.4401	0.4050	0.3540	0.3975	0.4105	0.4211
LR	0.5405	0.5120	0.4048	0.4548	0.4785	0.5124

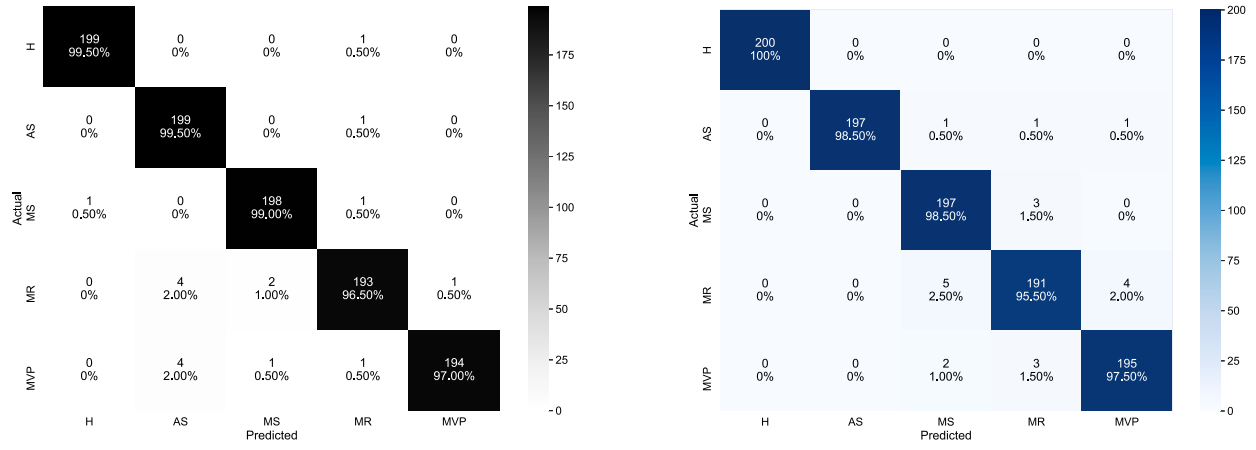
Table 10
Results in terms of Accuracy (Acc ± std), Sensitivity (Sens), Specificity (Spec), Precision (Prec), Recall (Rec), F1-score (F1) of other experiments on the Korean dataset for Health/Disease (H/D), Aortic/Mitral (A/M), and Mitral Stenosis/Mitral Regurgitation/Murmur in systole (MS/MR/MVP) classification tasks.

Classification tasks	Acc (±std)	Sens	Spec	Prec	Rec	F1
H/D	1.00	1.00	1.00	1.00	1.00	1.00
A/M	0.9950(±0.003)	–	–	1.00	0.98	0.98
$MS/MR/MVP$	0.9825(±0.012)	–	–	0.97	0.97	0.97

For both datasets, the same data preprocessing and feature extraction procedures described in Section 2.2.2 has been adopted. Afterwards, a comparison of the results obtained by Alkhodari and Fraiwan (2021) is performed. The experimental results of the five class tasks (H/AS/MS/MR/MVP) exploiting the Korean dataset are reported in Fig. 7. These results demonstrate that our approach performs slightly better than (Alkhodari & Fraiwan, 2021) for healthy subjects (100% vs. 99.50% accuracy) and those with MVP problems (97.50% vs. 97.00% of accuracy), but in the overall it has a slightly lower average accuracy (98.00% vs. 98.30%).

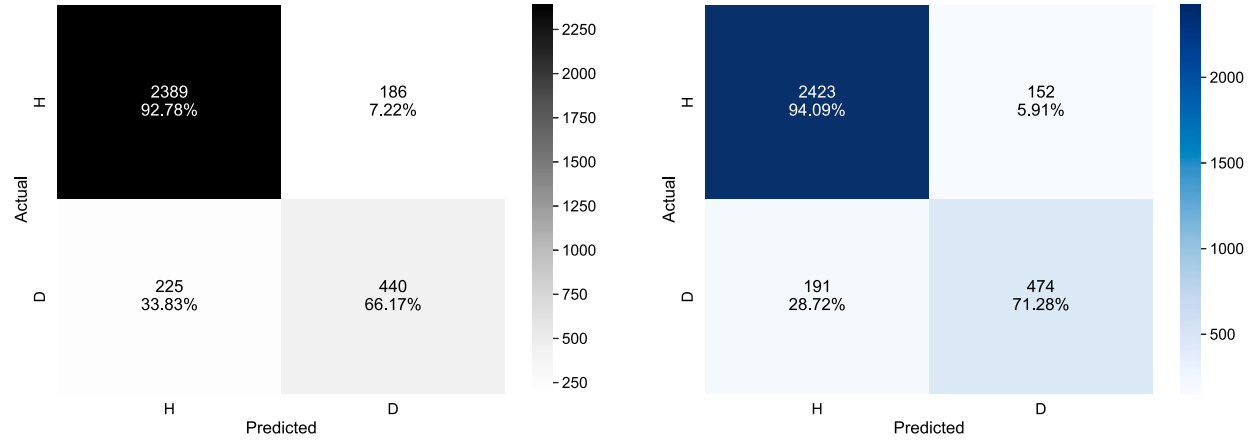
Fig. 8 shows the results of the Physionet dataset (Liu et al., 2016) which validates the proposed approach showing a higher accuracy (89.41% vs. 87.31%) than (Alkhodari & Fraiwan, 2021) in the binary (H/D) classification.

In addition, three extra experiments based on H/D, A/M, and MS/MR/MVP classification tasks (shown in Table 10) have been performed on the Korean dataset, obtaining the accuracy of 100.00%, 99.50%, and 98.25%, respectively, demonstrating the effectiveness of the proposed approach.



(a) (Alkhodari & Fraiwan, 2021) (b) Proposed approach

Fig. 7. Confusion matrices obtained by Alkhodari and Fraiwan (2021) (a) vs. the proposed approach (b) on the Korean dataset 5-class (H/AS/MS/MR/MVP) classification problem.



(a) (Alkhodari & Fraiwan, 2021) (b) Proposed approach

Fig. 8. Confusion matrices obtained by Alkhodari and Fraiwan (2021) (a) vs. the proposed approach (b) on the Physionet dataset binary (H/D) classification problem.

Table 11
Diagnosis tool Key Performance Indicators (KPI) mean values (on an I7-8GB laptop).

Benchmark	Mean service Time T_* (s)	Average CPU Utilization	RAM (Used in GB)	Storage (Audio file) + (Model)	Energy consumption (estimated mAh) (Murmuria, Medsger, Stavrou, & Voas, 2012)
Sound processing	0.348	0.1	–	–	0.0116
H/D	0.756	0.1	0.388	910 KB + 17.9 MB	0.0252
A/M	0.794	0.1	0.389	930 KB + 17.9 MB	0.0265
Mo/Se	0.784	0.1	0.389	915 KB + 17.9 MB	0.0261
St/Re	0.797	0.1	0.388	950 KB + 17.1 MB	0.0266
H/A/M	0.919	0.1	0.397	955 KB + 18.2 MB	0.0306
Model structure 3	2.695	0.1	1.554	3.70 MB + 70.8 MB	0.0898
Model structure 4	2.064	0.1	1.174	2.82 MB + 53.20 MB	0.0688
Model structure 5	1.901	0.1	1.554	3.70 MB + 70.8 MB	0.0634

6. Further insights

6.1. HVDD system feasibility study

As discussed in Section 2, the MCPS approach proposed in this paper aims to automatically diagnosing heart valve diseases, by implementing the HVDD framework of Fig. 1. Here, the technical feasibility of such an HVDD project, after the results and achievements provided by the training and learning efforts and experiments above described, is assessed. A first prototype of the HVDD framework, with only the light client and a server deployed on a local machine (an I7-8G laptop equipped with 8 GB of RAM) has been implemented as a proof-of-concept here for the experiments. The development of the full client features, the continuous learning component and the HVDDaaS deployment on a Cloud infrastructure, adopting serverless architecture, microservices and containers is ongoing.

Table 11 presents the mean values of the main Key Performance Indicators (KPIs) respectively for sound processing and for each classification level and model structure. After hundreds of experimental trials, we can argue that the HVDD system response time is in the order of few minutes, including the audio sampling auscultation (2–3 mins) and the time to diagnosis (2,22 s on average over the three model structures). The resource utilization measurements for a single diagnosis (also on

Table 12
ML model structure selection guidelines.

No of classes / dataset size	Small (2–4)	Medium (4–6)	Large (>6)
Small	M5, M3, M4, M2, M1	M5, M3, M4	M5
Medium	M3, M4, M5, M2, M1	M5, M3, M4, M2, M1	M5, M3, M4
Big	M2, M1, M3, M4, M5	M3, M4, M5, M2, M1	M5, M3, M4, M2, M1

the I7-8 GB laptop) show quite low values (CPU 0.1% and memory 1,42 GB on average, storage 68,34 MB and energy consumption 0,074 mAh overall). Such values demonstrate the feasibility of the proposed MCPS HVDD solution, allowing its deployment even in personal and mobile (i.e. resource constrained) devices. The same Table 11 provides the mean service time T_* (where the subscript $*$ has to be replaced by the benchmark initials) for inferring the corresponding pre-trained LSTM model taken as baseline, the average percentage of CPU utilization, the RAM used, the size of both the audio recording and the pre-trained model, and finally the energy consumption. Specifically, to estimate T_* on the three model structures the formulae specified by Eqs. (1), (2) and (3) have been exploited, respectively.

$$T_{M_3} = T_{SP} + T_{H/D} + T_{A/M} + \max(T_{St/Re}, T_{Mo/Se}) \quad (1)$$

$$T_{M_4} = T_{SP} + T_{H/A/M} + \max(T_{A/M}, T_{St/Re}, T_{Mo/Se}) \quad (2)$$

$$T_{M_5} = T_{SP} + T_{H/D} + \max(T_{A/M}, T_{St/Re}, T_{Mo/Se}). \quad (3)$$

In the *online mode*, which works online and thus requires a fully operating network connection, the subject data are immediately sent to the HVDDaaS Engine for processing, on-demand, as a service. The HVDDaaS Engine components process patient data by a specific HVD Classifier deployed and running on a virtual machine or a container instance on the Cloud. In both cases, all devices have to send patient data to the Cloud server for continuous learning. In the offline mode, data communication will take place when the network connection will be available, and therefore a specific local Patient DB is required as a temporary buffer.

6.2. Model structure selection guidelines

One of the most interesting contribution of this paper is represented by the combination of ML models into structure, a technique falling into the *ensemble learning* category (Sagi & Rokach, 2018). Based on the experience acquired from the different learning processes and experiments above discussed, here some simple guidelines and taxonomy for driving the selection process of the model structures depicted in Fig. 5 and described in Section 3 are specified. To such a purpose, the main criterion applied in such guidelines is based on the amount of data, the dataset size, or better the *data representativeness* (Tufekci, 2014) required by the classification tasks performed by the model structures.

It is possible to argue, as shown in Fig. 5, that the data representativeness requirement decreases from M1 to M5: the M1 classification task has the highest representativeness request, since the overall (DS_{All}) must have enough dataset items representing each of the considered classes, i.e. all classes (9 in the case study). M2 has similar data representativeness requirements due to the fact that, hierarchically splitting the original multiclass task into binary classification subtasks, reaches the same representativeness request of the M1 classification task at the lowest level (4) with its subtasks and corresponding datasets (DS_{DASt}, DS_{DARe}, DS_{DMSt}, DS_{DMRe}). Then, the M3 and M4 classification tasks have similar data representativeness requirements (at level 2 two binary classification problems on DS_{DA} and DS_{DM} datasets), lower than M1 and M2 ones. The lowest data representativeness requirements are for the level 2 classification tasks of M5, all based on the DS_D dataset. In brief, the representativeness requirements of a model can be represented by the function $R : \mathcal{M} \rightarrow \mathbb{R}^+$, where $\mathcal{M}$ is the model structure space. Thereby, an order relation of above models can be specified based on their representativeness as reported by Eq. (4)

$$R(M1) = R(M2) > R(M3) = R(M4) > R(M5). \quad (4)$$

The rationale behind the guidelines and taxonomy here proposed is based on the data representativeness requirements and relations of Eq. (4), considering the dataset size and the number of classes of the overall classification problem. In the case of the similar representativeness requirements, i.e. for M1–M2 and M3–M4, the accuracy can be taken into account as a further term of comparison, usually better in hierarchical models (M2 is better than M1 and M3 than M4 as also demonstrated in Table 9). Other parameters that can be taken into account can be the service time, the energy consumption, and/or the type of deployment (online or offline) as shown in Table 11.

Starting from all such criteria and parameters, Table 12 summarizes such (best) practice guidelines considering the number of classes, assuming a variable classification problem size, and the dataset size/data availability. The model lists there reported, are ranked based on their suitability to the problem at hand, i.e. the first in the list has the best suitability while the last the lowest suitability (e.g. for the M5, M3, M4, M2, M1 list M5 is the best and M1 is the worst). In the case of a small dataset, the best solution is the M5 model due to the low data representativeness requirements. If a low number of classes is considered in the overall classification problem (e.g. when the goal is to classify between healthy and diseased or to identify the valve disease between aortic or mitral for diseased patients), however, the number of levels decreases and even a flat multiclass classification problem and structure such as M1 can be adopted with a small dataset. When the size of the dataset increases, model structures with higher data representativeness requirements can be taken into account, also considering other parameters such as accuracy, time and deployment/resources. In Table 12 representativeness-accuracy criteria are enforced, resulting in different model suitability orders. This study has some *limitations*. *First*, patient-relevant informations are limited. Because of data protection regulations, all phonocardiography data used in this study were anonymized and stripped of identifying meta data. Therefore, we were not able to maintain subject-level independence for the training–validation–testing splits. Also, further analyses on age, sex, and relevant clinical informations were not possible. If these data would become available in the future, the current analyses could be extended to investigate the possibility of integration of other patient data in the model. *Second*, availability of other public data sets with diagnostic labels are needed to further verify the generalization capabilities of our models. *Third*, the misclassification rates of the models, particularly for the mild classes, were still high. Failing to identify the mild cases might have important clinical implications, such as delayed diagnosis and treatment. Therefore, further refinement of the models to decrease the misclassification rates is needed before their deployment in clinical routine.

7. Conclusions

In this paper, we presented an innovative and intelligent MCPS for real-time processing of heartbeat audio files using ML models. The system extends the automatic heart valve disease classification to 9 classes through hierarchical ML algorithms using a dataset of 132 subjects collected and made available as a contribution of this paper.

Various ML models and integration methods were proposed and validated, achieving an accuracy of over 99% in differentiating health and disease states. However, the system does not replace medical doctors in final diagnoses. It offers an automated diagnostic tool deployable on personal devices with pre-trained models and digital stethoscopes. The proposed framework is quite effective and promising for patients, healthcare practitioners, and researchers, considering its impact on the diagnostic process. It can be used in different ways: (i) to support the doctor in the diagnosis, speeding up the process; (ii) as a pre-screening tool for nurses and/or receptionists to assess and assign the degree of urgency (triage); (iii) for the patient self-diagnosis/assessment (at home with her personal device); (iv) in emergency, remote or uncomfortable conditions. To demonstrate the feasibility of the HVDD MCPS an investigation on resource utilization has been performed, also providing further insights and guidelines on how to use the hierarchical models, including threats to validity.

CRediT authorship contribution statement

Gennaro Tartarisco: Conceptualization, Methodology, Software, Writing. **Giovanni Cicceri:** Conceptualization, Methodology, Software, Writing. **Roberta Bruschetta:** Conceptualization, Methodology, Supervision, Validation. **Alessandro Tonacci:** Conceptualization, Validation, Resources. **Simona Campisi:** Conceptualization, Validation, Resources. **Salvatore Vitabile:** Conceptualization, Validation, Resources. **Antonio Cerasa:** Conceptualization, Methodology, Resources, Validation, Supervision, Writing. **Salvatore Distefano:** Conceptualization, Methodology, Software, Supervision, Writing, Project administration. **Alessio Pellegrino:** Conceptualization, Methodology, Resources, Validation, Supervision, Writing. **Pietro Amedeo Modesti:** Conceptualization, Methodology, Software, Supervision, Writing, Project administration. **Giovanni Pioggia:** Conceptualization, Methodology, Software, Supervision, Writing, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Alkhodari, M., & Fraiwan, L. (2021). Convolutional and recurrent neural networks for the detection of valvular heart diseases in phonocardiogram recordings. *Computer Methods and Programs in Biomedicine*, *200*, Article 105940.

Association, A. H., et al. (2017). Cardiovascular disease costs will exceed $1 trillion by 2035, warns the American Heart Association. *Internet Document*, *14*.

Austin, P. C., Tu, J. V., Ho, J. E., Levy, D., & Lee, D. S. (2013). Using methods from the data-mining and machine-learning literature for disease classification and prediction: a case study examining classification of heart failure subtypes. *Journal of Clinical Epidemiology*, *66*(4), 398–407.

Bengio, Y., & Grandvalet, Y. (2003). No unbiased estimator of the variance of k-fold cross-validation. *Advances in Neural Information Processing Systems*, *16*.

Bhandare, D., Patel, D., & Shimshak, T. (2022). Portable artificial intelligence device detects heart murmurs in real time in tandem with a regular stethoscope. *Journal of the American College of Cardiology*, *79*(9_Supplement), 2005.

Bruneo, D., & De Vita, F. (2019). On the use of LSTM networks for predictive maintenance in smart industries. In *2019 IEEE international conference on smart computing (SMARTCOMP)* (pp. 241–248). IEEE.

Chizner, M. A. (2008). Cardiac auscultation: rediscovering the lost art. *Current Problems in Cardiology*, *33*(7), 326–408.

Chollet, F., et al. (2018). Keras: The python deep learning library. *Astrophysics Source Code Library*, ascl–1806.

Clifford, G. D., Liu, C., Moody, B. E., Roig, J. M., Schmidt, S. E., Li, Q., et al. (2017). Recent advances in heart sound analysis. *Physiological Measurement*, *38*, E10–E25.

Coffey, S., Roberts-Thomson, R., Brown, A., Carapetis, J., Chen, M., Enriquez-Sarano, M., et al. (2021). Global epidemiology of valvular heart disease. *Nature Reviews Cardiology*, *18*(12), 853–864.

Cogswell, M., Ahmed, F., Girshick, R., Zitnick, L., & Batra, D. (2015). Reducing overfitting in deep networks by decorrelating representations. arXiv preprint arXiv:1511.06068.

Darmawahyuni, A., Nurmaini, S., & Firdaus, F. (2019). Coronary heart disease interpretation based on deep neural network. *Computer Engineering and Applications Journal*, *8*(1), 1–12.

Dong, F., Qian, K., Ren, Z., Baird, A., Li, X., Dai, Z., et al. (2019). Machine listening for heart status monitoring: Introducing and benchmarking hss—the heart sounds shenzhen corpus. *IEEE Journal of Biomedical and Health Informatics*, *24*(7), 2082–2092.

Dwivedi, A. K., Imtiaz, S. A., & Rodriguez-Villegas, E. (2018). Algorithms for automatic analysis and classification of heart sounds–a systematic review. *IEEE Access*, *7*, 8316–8345.

Edgar, T. W., & Manz, D. O. (2017). Exploratory study. *Research Methods for Cyber Security*, *29*, 95–130.

El Badlaoui, O., & Hammouch, A. (2017). Phonocardiogram classification based on MFCC extraction. In *2017 IEEE international conference on computational intelligence and virtual environments for measurement systems and applications (CIVEMSA)* (pp. 217–221). IEEE.

Gokmen, T., Rasch, M. J., & Haensch, W. (2018). Training LSTM networks with resistive cross-point devices. *Frontiers in Neuroscience*, *12*, 745.

Gokulnath, C. B., & Shantharajah, S. (2019). An optimized feature selection based on genetic approach and support vector machine for heart disease. *Cluster Computing*, *22*(6), 14777–14787.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Khemphila, A., & Boonjing, V. (2010). Comparing performances of logistic regression, decision trees, and neural networks for classifying heart disease patients. In *2010 international conference on computer information systems and industrial management applications (CISIM)* (pp. 193–198). IEEE.

Liu, C., Springer, D., Li, Q., Moody, B., Juan, R. A., Chorro, F. J., et al. (2016). An open access database for the evaluation of heart sound algorithms. *Physiological Measurement*, *37*(12), 2181.

Markandey, V. (2010). *ECG implementation on the TMS320C5515 DSP medical development kit (MDK): Texas instruments application report jun, 35*.

Masetic, Z., & Subasi, A. (2016). Congestive heart failure detection using random forest classifier. *Computer Methods and Programs in Biomedicine*, *130*, 54–64.

McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., et al. (2015). Librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference, Vol. 8* (pp. 18–25).

Murmuria, R., Medsger, J., Stavrou, A., & Voas, J. M. (2012). Mobile application and device power usage measurements. In *2012 IEEE sixth international conference on software security and reliability* (pp. 147–156). IEEE.

Pal, M. (2005). Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, *26*(1), 217–222.

Pascanu, R., Gulcehre, C., Cho, K., & Bengio, Y. (2013). How to construct deep recurrent neural networks. arXiv preprint arXiv:1312.6026.

Pattekari, S. A., & Parveen, A. (2012). Prediction system for heart disease using Naïve Bayes. *International Journal of Advanced Computer and Mathematical Sciences*, *3*(3), 290–294.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Roth, G. A., Mensah, G. A., Johnson, C. O., Addolorato, G., Ammirati, E., Baddour, L. M., et al. (2020). Global burden of cardiovascular diseases and risk factors, 1990–2019: update from the GBD 2019 study. *Journal of the American College of Cardiology*, *76*(25), 2982–3021.

Rubin, J., Abreu, R., Ganguli, A., Nelaturi, S., Matei, I., & Sricharan, K. (2016). Classifying heart sound recordings using deep convolutional neural networks and mel-frequency cepstral coefficients. In *2016 Computing in Cardiology Conference (CinC)* (pp. 813–816). IEEE.

Sagi, O., & Rokach, L. (2018). Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(4), Article e1249.

Sethi, Y., Patel, N., Kaka, N., Desai, A., Kaiwan, O., Sheth, M., et al. (2022). Artificial intelligence in pediatric cardiology: A scoping review. *Journal of Clinical Medicine*, *11*(23), 7072.

Shanthi, T., Anand, R., Annapoorani, S., & Birundha, N. (2023). Analysis of phonocardiogram signal using deep learning. In *International conference on innovative computing and communications* (pp. 621–629). Springer.

Smagulova, K., & James, A. P. (2019). A survey on LSTM memristive neural network architectures and applications. *The European Physical Journal Special Topics*, *228*(10), 2313–2324.

Son, G.-Y., & Kwon, S. (2018). Classification of heart sound signal using multiple features. *Applied Sciences*, *8*(12), 2344.

Subbalakshmi, G., Ramesh, K., & Rao, M. C. (2011). Decision support in heart disease prediction system using naive bayes. *Indian Journal of Computer Science and Engineering (IJCSE)*, *2*(2), 170–176.

Tian, Y., & Zhang, Y. (2022). A comprehensive survey on regularization strategies in machine learning. *Information Fusion*, *80*, 146–166.

Tufekci, Z. (2014). Big questions for social media big data: Representativeness, validity and other methodological pitfalls. In *Eighth international AAAI conference on weblogs and social media*.

Uçar, M. K., Nour, M., Sindi, H., & Polat, K. (2020). The effect of training and testing process on machine learning in biomedical datasets. *Mathematical Problems in Engineering*, *2020*.

Usman, M., Ahmad, Z., & Wajid, M. (2019). Dataset of raw and pre-processed speech signals, Mel Frequency Cepstral Coefficients of Speech and Heart Rate measurements. In *2019 5th international conference on signal processing, computing and control (ISPCC)* (pp. 376–379). IEEE.

Varshney, L. R., & Sun, J. Z. (2013). Why do we perceive logarithmically? *Significance*, *10*(1), 28–31.

Vasudevan, R. S., Horiuchi, Y., Torriani, F. J., Cotter, B., Maisel, S. M., Dadwal, S. S., et al. (2020). Persistent value of the stethoscope in the age of COVID-19. *The American Journal of Medicine*, *133*(10), 1143–1150.

Watkins, D. A., Johnson, C. O., Colquhoun, S. M., Karthikeyan, G., Beaton, A., Bukhman, G., et al. (2017). Global, regional, and national burden of rheumatic heart disease, 1990–2015. *New England Journal of Medicine*, *377*(8), 713–722.