ENHANCED COMPUTATIONAL METHODS FOR DETECTION AND INTERPRETATION OF HEART DISEASE BASED ON ENSEMBLE-LEARNING AND AUTOENCODER FRAMEWORK

ABDALLAH OSAMA HAMDAN ABDELLATIF

FACULTY OF ENGINEERING
UNIVERSITI MALAYA
KUALA LUMPUR

2024

ENHANCED COMPUTATIONAL METHODS FOR DETECTION AND INTERPRETATION OF HEART DISEASE BASED ON ENSEMBLE-LEARNING AND AUTOENCODER FRAMEWORK

ABDALLAH OSAMA HAMDAN ABDELLATIF

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF ENGINEERING
UNIVERSITI MALAYA
KUALA LUMPUR

2024

UNIVERSITI MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Abdallah Osama Hamdan Abdellatif.

Matric No: 17221028

Name of Degree: Doctor of Philosophy

Title of Dissertation: Enhanced Computational Methods for Detection and Interpretation of Heart Disease Based on Ensemble-Learning and Autoencoder Framework.

Field of Study: Control System (Engineering and Engineering Trades)

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work.
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the Universiti Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work, I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date: 5-9-2024

Subscribed and solemnly declared before,

Witness's Signature Date: 5-9-2024

Name:

Designation:

ENHANCED COMPUTATIONAL METHODS FOR DETECTION AND INTERPRETATION OF HEART DISEASE BASED ON ENSEMBLE-LEARNING AND AUTOENCODER FRAMEWORK.

ABSTRACT

Heart disease remains the primary cause of mortality globally, and its early detection is critical for reducing mortality rates. However, the challenge of class imbalance and high dimensionality in clinical data significantly impedes the efficacy of Machine Learning (ML) models in this domain. This thesis presents two innovative methods that holistically address these challenges at algorithmic and data levels to enhance heart disease detection. The first method introduces an Improved Weighted Random Forest (IWRF) approach, focusing on algorithmic innovation to tackle the imbalance problem. It employs supervised infinite feature selection (Inf-FSs) to identify significant features and Bayesian optimization for fine-tuning hyperparameters. Validated on Statlog and heart disease clinical records datasets, this method demonstrates a notable improvement in prediction accuracy and F-measure, outperforming existing models and marking an accuracy enhancement of 2.4% and 4.6% on these datasets. In contrast, the second method addresses the data-level imbalance through a novel framework named Conditional Autoencoder with Stack Predictor for Heart Disease (CAVE-SPFHD). This approach integrates a conditional variational autoencoder (CVAE) to effectively balance the dataset and a stack predictor (SPFHD) that utilizes tree-based ensemble learning algorithms. The base models' predictions are integrated using a support vector machine, significantly enhancing detection accuracy. Tested across four datasets, CAVE-SPFHD surpasses state-of-the-art methods in f1-score, providing improved not only predictive performance but also critical interpretative insights using the SHapley Additive explanation (SHAP) algorithm. Together, these two methods represent a comprehensive approach to heart disease detection in ML, effectively addressing the dual challenges of class imbalance

and high dimensionality. By innovatively tackling these issues at both the algorithm and data levels, this thesis significantly contributes to the field, offering robust, accurate, and interpretable ML solutions for early heart disease detection, which is crucial for proactive healthcare interventions.

Keywords: Heart disease, Conditional variational auto-encoder, Stacking ensemble learning, SHAP, Tree ensemble, Hyperparameter optimization, Feature selection, Imbalance.

KAEDAH KOMPUTASIONAL YANG DIPERTINGKATKAN UNTUK PENGESANAN DAN TAFSIRAN PENYAKIT JANTUNG BERDASARKAN PEMBELAJARAN ENSEMBLE DAN RANGKA KERJA AUTOENCODER..

ABSTRAK

Penyakit jantung merupakan penyebab utama kematian di seluruh dunia, dan pengesanan awalnya adalah kritikal untuk mengurangkan kadar mortaliti. Namun, cabaran ketidakseimbangan kelas dan dimensi data yang tinggi secara signifikan menghalang keberkesanan model Pembelajaran Mesin (ML) dalam domain ini. Tesis ini mempersembahkan dua kaedah inovatif yang secara holistik menangani cabaran ini pada kedua-dua tahap algoritma dan data untuk meningkatkan pengesanan penyakit jantung. Kaedah pertama memperkenalkan pendekatan Random Forest Berbobot Terbaik (IWRF), yang memberi tumpuan pada inovasi algoritma untuk menangani masalah ketidakseimbangan. Ia menggunakan pemilihan ciri tak terhingga yang diawasi (Inf-FSs) untuk mengenal pasti ciri-ciri penting dan pengoptimuman Bayesian untuk penyelarasan halus hiperparameter. Divalidasi pada dataset Statlog dan rekod klinikal penyakit jantung, kaedah ini menunjukkan peningkatan yang ketara dalam ketepatan ramalan dan ukuran F, mengatasi model-model sedia ada dan mencatatkan peningkatan ketepatan sebanyak 2.4% dan 4.6% pada kedua-dua dataset tersebut. Sebaliknya, kaedah kedua menangani ketidakseimbangan data melalui kerangka baru yang dinamakan Autoencoder Bersyarat dengan Penumpu Ramal untuk Penyakit Jantung (CAVE-SPFHD). Pendekatan ini mengintegrasikan autoencoder variasi bersyarat (CVAE) untuk menyeimbangkan dataset secara efektif, digabungkan dengan penumpu ramal (SPFHD) yang menggunakan algoritma pembelajaran ansambel berasaskan pohon. Ramalan dari model asas diintegrasikan menggunakan mesin vektor sokongan, meningkatkan ketepatan pengesanan secara signifikan. Diuji merentas empat dataset, CAVE-SPFHD mengatasi kaedah terkini dalam skor f1, bukan sahaja memberikan prestasi ramalan yang lebih baik

tetapi juga wawasan interpretatif kritikal menggunakan algoritma Penjelasan Aditif SHapley (SHAP). Secara bersama-sama, kedua-dua kaedah ini mewakili pendekatan menyeluruh untuk pengesanan penyakit jantung dalam ML, menangani cabaran ketidakseimbangan kelas dan dimensi data secara efektif. Dengan menangani isu-isu ini secara inovatif pada kedua-dua tahap algoritma dan data, tesis ini memberikan sumbangan yang signifikan ke dalam bidang ini, menawarkan penyelesaian ML yang kuat, tepat, dan boleh diinterpretasi untuk pengesanan awal penyakit jantung, yang penting untuk intervensi kesihatan proaktif.

Kata kunci: Penyakit Jantung, Auto-Encoder Variasi Bersyarat, Pembelajaran Ansambel Bertingkat, SHAP, Ansambel Pohon, Pengoptimuman Hiperparameter, Pemilihan Ciri, Ketidakseimbangan

ACKNOWLEDGEMENTS

I am grateful to Allah SWT for making this journey easy for me. I am thankful to all those people who helped me in completing this research project.

I would also like to express immense amount of gratitude to my supervisor, Associate prof. Ir. Dr. Jeevan A/L Kanesan for his professional advice and helpful comments related to the project. Throughout the project, Dr. Jeevan consistently shared his ideas and suggestions in order to improve the quality of research work.

I would also like to thank my co-supervisors, Associate prof. Ir. Dr. Chow Chee Onn

Associate prof. Ir. Dr. Chuah Joon Huang for all their great effort in guiding me and advising me. Their patience has been phenomenal during this project.

I would also like to thank parents for their consistent prayers, moral and financial support. I would also express my deepest gratitude to my brother, for without his constant support I would not have been able to accomplish this task.

I would also like to thank my colleagues for their support and encouragement throughout this work. I am thankful to Allah SWT for blessing me with friends, who helped me numerous times with their valuable insights throughout this journey.

TABLE OF CONTENTS

Abstract iii

Abstrak v

Acknowledgements vii

Table of Contents viii

List of Figures xii

List of Tables xiii

CHAPTER 1: INTRODUCTION 1

1.1 Background 1

1.2 Problem Statement 3

1.2.1 Problem Due to Class Imbalance Dataset 4

1.2.2 Problem in Existing Machine Learning Models and Optimization Techniques in CVD Predictive Systems 5

1.2.3 Problem-related to ML model interpretation 6

1.3 Research Objectives 7

1.4 Research Methodology 8

1.5 Scope of Research 9

1.6 Research Outline 10

CHAPTER 2: LITERATURE REVIEW 11

2.1 Introduction 11

2.2 An overview of heart disease 11

2.3 Data Preprocessing 12

2.3.1 Data cleaning and transformation 12

2.3.2 Feature importance 13

2.4 Data balancing 14
2.4.1 Random Over-sampling 15
2.4.2 Synthetic Minority Over-sampling Technique (SMOTE) 17
2.5 Machine learning algorithms 18
2.5.1 Supervised learning 19
2.5.1.1 Decision tree algorithm 20
2.5.1.2 Support vector machine 21
2.5.1.3 k-Nearest Neighbors 23
2.5.1.4 Naïve Bayes Classifier 24
2.5.1.5 Logistic regression 24
2.5.1.6 Random Forest 25
2.5.2 Unsupervised learning 26
2.5.2.1 Clustering 27
2.5.2.2 Principal component analysis 28
2.6 Deep Learning 28
2.6.1 Convolutional Neural Network 29
2.6.2 Recurrent Neural Network 29
2.7 Application on heart disease detection using machine learning. 30
2.7.1 Applications of Logistic Regression 30
2.7.2 Applications of Support vector machine 31
2.7.3 Applications of k-Nearest Neighbors 32
2.7.4 Applications of naïve Bayes 32
2.7.5 Applications of Decision Tree 34
2.8 Applications of Deep learning 35
2.8.1 Applications of Convolutional Neural Network 36
2.8.2 Applications of Recurrent Neural Network 37

2.8.3 Applications of Artificial Neural Network 38
2.9 Applications of Data balancing 39
2.10 Applications of tuning the hyperparameter optimization 40
2.11 Summary 41

CHAPTER 3: METHODOLOGY 47

3.1 Introduction 47
3.2 Data Collection and preprocessing 47
3.2.1 Cleveland dataset 49
3.2.2 Statlog 50
3.2.3 Z-Alizadeh Sani 51
3.2.4 Heart disease clinical records 55
3.3 Proposed Method for CVD detection and tackling the imbalanced issue on the algorithm level. 57
3.3.1 Feature selection using infinite feature selection. 58
3.3.2 Improved Weighted Random Forest 60
3.3.3 Bayesian Optimization 65
3.3.4 Particle Swarm Optimization 71
3.3.5 Genetic Algorithm 75
3.4 Proposed Method for CVD detection and tackling the imbalanced issue on the data level..... 80
3.4.1 CVAE-based method for data balancing 81
3.4.2 SPFHD Framework 84
3.5 Performance evaluation metrics 87
3.6 Model interpretation using SHapley Additive exPlanations (SHAP) 88
3.7 Summary 92

CHAPTER 4: RESULT AND DISCUSSION 94

4.1 The proposed Inf-FSs-IWRF model. 94

4.2 Feature Selection Results for the proposed Inf-FSs-IWRF model. 94

4.2.1 Classification results for the proposed Inf-FSs-IWRF model. 97

4.3 The proposed SPFHD model 102

4.3.1 Classification results for the proposed SPFHD model 102

4.3.2 Hyperparameter optimization results for the proposed SPFHD model .. 106

4.3.3 Model interpretation for the proposed SPFHD model 108

4.4 Comparative Analysis 119

CHAPTER 5: CONCLUSION AND FUTURE WORK 125

References 130

LIST OF FIGURES

Figure 2.1: Machine learning classification models. 19

Figure 3.1: Flowchart of the proposed method one (IWRF)..... 58

Figure 3.2: BO-TPE A Graphical Representation of Bayesian Hyperparameter Tuning with Tree Parzen Estimator of ML Models..... 71

Figure 3.3: PSO Workflow for Hyperparameter Tuning of ML Models..... 75

Figure 3.4: GA Workflow for Hyperparameter Tuning of ML Models 79

Figure 3.5: Flowchart of the proposed method two (SPFHD)..... 80

Figure 3.6: The difference between the information flow of a VAE and conventional AE 82

Figure 3.7: Schematic diagram of the functioning of the proposed CVAE model..... 83

Figure 3.8: The entire procedure for the SPFHD model during (a) training and (b) testing. 87

Figure 3.9: The implementation of the SHAP framework in interpreting the SPFHD model..... 92

Figure 4.1: The comparison between the proposed IWRF and SMOTE-RF..... 102

Figure 4.2: features rank based on SHAP values for SPFHD prediction of presence and patient survival of CVD. Red indicates a high value, blue is a low value for attributes, whereas SHAP values (negative or positive) reflect the directionality of the attributes. Negative SHAP values represent negative predictions (absence (a, b, c) and alive (d)). Conversely, positive SHAP values signify positive predictions (presence (a, b, c) and death event (d)). 113

Figure 4.3: The effect of each attribute on SPFHD's output according to the SHAP framework where each figure refers to the ranking of the attribute effectiveness on different datasets (a) Cleveland, (b) Statlog, (c) Z-AliZadeh Sani, and (d) HD Clinical Records..... 116

LIST OF TABLES

Table 2.1: Summary of Literature Review.....43

Table 3.1: Summary of Features in the Cleveland and Statlog Heart Disease Datasets: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature. 50

Table 3.2: Summary of Features in the Z-Alizadeh Heart Disease Dataset: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature. 52

Table 3.3: Summary of Features in the Heart Disease Clinical Records Dataset: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature. 56

Table 3.4: The summary of the utilized datasets in the study.....57

Table 3.5: Model Hyperparameter Exploration: A Comprehensive Table of Hyperparameters Selections, Types, and Ranges 70

Table 4.1: Feature ranking and weight importance determined by Inf-FS_s 96

Table 4.2: Selected features of both datasets.....97

Table 4.3: Performance evaluation of Statlog dataset for the proposed FS method.....99

Table 4.4: Performance evaluation of HD clinical records dataset for the proposed FS method.....100

Table 4.5: Comparison results between IWRF and SMOTE-RF on Statlog dataset 101

Table 4.6: Comparison results between IWRF and SMOTE-RF on HD clinical record dataset.....101

Table 4.7: The results for the SPFHD and the base models on the balanced datasets using the default hyperparameters. 103

Table 4.8: The results for the SPFHD with different data balancing methods on two datasets 104

Table 4.9: The results for the SPFHD and base learners with CVAE-based method for data balancing methods on two datasets 105

Table 4.10: The results for the SPFHD with different HPO methods on different datasets 107

Table 4.11: The selected features after applying the SHAP analysis for the four datasets. 116

Table 4.12: The comparison between the proposed model with optimal features selected by SHAP. 117

Table 4.13: The results of the Friedman rank and Iman-Davenport tests, as well as the mean value of the accuracy (%). 119

Table 4.14: Comparison of the proposed method and other classifiers with respect to Friedman's post hoc test 119

Table 4.15: Performance comparison between the proposed method and previous work on the Cleveland dataset. 121

Table 4.16: Performance comparison between the proposed method and previous work on the Statlog dataset. 122

Table 4.17: Performance comparison between the proposed method and previous work on the Z-Alizadeh Sani dataset. 123

Table 4.18: Performance comparison between the proposed method and previous work on the HD clinical records dataset. 124

LIST OF SYMBOLS AND ABBREVIATIONS

AdaBoost	:	Adaptive Boosting
AF	:	Atrial Fibrillation
AI	:	Artificial Intelligence
ANN	:	Artificial Neural Network
AUC	:	Area Under the Curve
BO	:	Bayesian Optimization
CART	:	Classification And Regression Tree
CNN	:	Convolutional Neural Network
CVAE	:	Conditional Variational Autoencoder
CVD	:	Cardiovascular Disease
DL	:	Deep Learning
DT	:	Decision Tree
ECG	:	Electrocardiogram
ETC	:	Extra Tree Classifier
FS	:	Feature Selection
GA	:	Genetic Algorithm
GP	:	Gaussian Process
GS	:	Grid Search
HF	:	Heart Failure
HNB	:	Hidden Naïve Bayes
HP	:	Hyperparameter
HPO	:	Hyperparameters Optimization
ICU	:	Intensive Care Unit

k-NN	:	K-Nearest Neighbors
LVAD	:	Left Ventricular Assist Device
ML	:	Machine Learning
NB	:	Naïve Bayes
NN	:	Neural Networks
PCA	:	Principal Component Analysis
PSO	:	Particle Swarm Optimization
RFC	:	Random Forest Classifier
RFR	:	Random Forest Regression
RNN	:	Recurrent Neural Network
SMOTE	:	Synthetic Minority Oversampling Technique
SPFHD		A Stack Predictor for Heart Disease
STFT	:	Short-Term Fourier Transform
SVM	:	Support Vector Machine
SWT	:	Stationary Wavelet Transform
TPE	:	Tree-Structured Parzen Estimator
WHO	:	World Health Organization
p	:	Probability
w	:	Weight Vector
b	:	Bias
k	:	Number of Neighbors
μ	:	Mean
σ	:	Standard Deviation
m	:	Mmutual Information
h	:	Fisher Criteria

A	:	Adjacency Matrix
M	:	Number of Trees
I	:	Indicator Function
α	:	Weighting Factors
ε	:	Error
D	:	Search Space
l	:	Good Outcome
g	:	Bad Outcome
c1	:	Congnitive Factor
c2	:	Social Factor
r	:	Random Numbers
Pbest	:	Best Known Position
Gbest	:	Best Known Global Position
t	:	Iteration Number
γ_i	:	SHAP Values
C	:	Feature Subset

CHAPTER 1: INTRODUCTION

1.1 Background

Heart disease is a cardiovascular disease (CVD) that persists as the leading cause of death worldwide and accounts for roughly 30 percent of global deaths (Rana et al., 2021). It kills more individuals than any other cause. For example, in 2016, the death rate was approximately 17.9 million, accounting for a third of global mortality; 85% of the total died due to a heart attack or a stroke. Based on the World Health Organization (WHO) projections, the worldwide death toll is expected to reach approximately 23.6 million by 2030 if no action is taken. In Malaysia, the burden of CVD death and morbidity has increased during the past three decades. The Malaysian Ministry of Health reported that cardiovascular disease remained the major cause of mortality from the 1980s to the present. For example, in 2017, it is anticipated that chronic diseases, including CVD, diabetes, and cancer, incurred a total of RM 9.65 billion in actual medical expenses. Improving the early detection and treatment of CVD would benefit Malaysia's and the global economy's public health (Benjamin et al., 2019). Heart disease remains the main culprit in cutting short the lives of Malaysians—it’s the number one cause of premature deaths in the country. Premature deaths refer to lives lost between the ages of 30 and 69, below Malaysia's average life expectancy of around 75 years. A total of 95,266 deaths were recorded in 2022 within this age group, according to the Statistics on Causes of Death Malaysia 2023 report. Heart disease was the top killer, accounting for 18.4% of medically certified deaths in that year, based on the report by the Department of Statistics Malaysia (Meikeng, 2023).

On a fundamental level, clinical medicine works by detecting the symptoms and signs of disease in patients who present themselves. The physician collects a patient's medical history and vitals and performs a physical examination. This information is utilized to establish a list of potential diagnoses, further refined by laboratory tests, diagnostic

procedures, or imaging as needed. These data are evaluated and used to prescribe the proper behavioural adjustments, drugs, therapeutic procedures, or surgical procedures. This paradigm has remained essentially unchanged for decades. However, evidence for the efficacy of novel diagnostics and data sources such as whole-genome sequencing, pharmacogenomics, and mobile device data continues to accumulate. This trend is predicted to increase exponentially, particularly as the cost of –omics research, computation decreases, and wearable devices (Tsao et al., 2022).

Moreover, linkages between CVD illness and inflammatory, neurological, and other chronic disorders are becoming increasingly apparent and will eventually play a significant role in the practice of CVD medicine (Arabasadi et al., 2017; Tsao et al., 2022). This will require cardiologists to analyze and implement knowledge from other biomedical domains. Concurrently, physicians are spending less time with each patient, and patients are demanding more quick and individualized care. In fact, physicians are overwhelmed with data that necessitates a more complex interpretation while also being asked to perform more efficiently. In cardiology, the promise of artificial intelligence (AI) and machine learning (ML) is to give the required tools to supplement and expand the cardiologist's function. With these technologies, every step of the patient care process might be improved, from the initial diagnosis to the selection and monitoring of medicines using real-time, companion diagnostics. Therefore, an accurate diagnosis is vital, and good treatment can lower the likelihood of illness progression. To enhance the diagnosis, a thorough knowledge of the risk is necessary (Tsao et al., 2022).

The conventional method diagnoses disorders by assessing a patient's symptoms and medical history, such as an electrocardiogram (ECG) testing, blood glucose levels, blood pressure, and cholesterol levels. However, this procedure is time-intensive and costly. It is simplified with the use of ML. This methodology saves a great deal of time and, as a

result, enhances the effectiveness of the diagnosis, especially with the availability of clinical data and patients' medical histories (Beunza et al., 2019). The amount of data available increases daily, and hospitals gradually embrace big data technologies (Pramanik et al., 2022). Utilizing clinical data in the health establishment yields enormous advantages, such as enhancing the findings and minimizing expenses. Effective deployment of ML boosts the efficiency and effectiveness of healthcare services. Applying ML has shown considerable improvement in clinical data diagnosis. For example, diabetes, CVD, and breast cancer have been diagnosed utilizing ML (Weng et al., 2017).

1.2 Problem Statement

Although the performance of CVD detection is exceptional, due to various reasons such as outliers, noise, high dimensional features of CVD patients and the class imbalance problem among classes, the overall performance and CVD detection accuracy are significantly degraded. The high-dimensional space includes redundant and irrelevant characteristics, both of which lower CVD detection accuracy and increase the ML model complexity and computational time. Moreover, real-world data is not simply high-dimensional since there are several intrinsic features, which are the features that create the observed class, that must be considered. Sometimes it is unclear precisely what should be assessed, which could result in repeated measurements of traits that are combinations of other attributes. CVD patients (minority class) and non-CVD (majority class) comprise the two classes. The proportion between the two classes varies, with various proportions between the two classes in different datasets. For example, according to the heart disease clinical records (Ahmad et al., 2017), 67.55% of the individuals survived, and 32.44% were deceased for 299 CVD patients. Further, in some cases, the minor and major classes change where non-CVD individuals are the minor cases, and the CVD patients are a major

class. For example, in Z-Alizadeh Sani (Arabasadi et al., 2017), 71.28% of the individuals have CVD, while the 28.71% are non-CVD.

Individuals belonging to a majority class are likely to have a greater variation in the number of individuals compared to those of a minority class. The majority of samples significantly impact the training process; hence, the trained machine learning model is biassed and tends to classify data as belonging to the majority class (2017). This discrepancy in sample quantities between the two groups leads to the incorrect identification of CVD patients during testing, and the cumulative loss exceeds the final loss. The combination of high-dimension features and imbalanced data reduces the effectiveness of the trained ML model and renders the learning process ineffective for distinguishing CVD connections from the data (Blagus & Lusa, 2013; Sağlam & Cengiz, 2022). Therefore, it becomes a scheming task to balance non-CVD and CVD individuals.

1.2.1 Problem Due to Class Imbalance Dataset

The clinical data is categorized into two classes, i.e., CVD patients (minority class) and non-CVD individuals (majority class). The non-CVD individual samples have a higher number of samples than the CVD individuals. The samples corresponding to a majority class tend to have more samples compared to minority class samples. Therefore, the majority class samples influence the training process; the trained ML model becomes biased and tends to classify the samples as the majority class (Blagus & Lusa, 2013). This variation in sample numbers between two classes leads to the false detection of a CVD patient in the testing process, and the cumulative loss overwhelms the final loss. The high-dimensional features and imbalanced data decrease the trained model's performance and make the learning process inefficient in distinguishing clinical relationships from the data (2019). it becomes a scheming task to balance non-CVD and CVD individuals. For instance, the Synthetic Minority Oversampling Technique (SMOTE) is widely utilized in many works (Umer et al., 2022). In (Ishaq et al., 2021), the Extra Tree Classifier (ETC)

was proposed where the Random Forest Classifier (RFC) is used for the feature selection (FS), and the SMOTE is utilized to make the data balances. SMOTE-based artificial neural network (ANN) was mentioned by (Waqar et al., 2021). In contrast, the Randomoversampler was employed for data balancing (Kibria & Matin, 2022), and the ML-based fusion approach consisting of adaptive boosting (AdaBoost) combined with a decision trees model was proposed for heart disease severity prediction. Also, [30] employed the RFC to predict heart disease (HD) using the SOMTE for dataset balancing.

Most of the prior studies employed SMOTE method to balance the data distribution. However, SMOTE method has some disadvantages. First, the newly generated samples might fall in the majority class region, causing overlapping and generating noise patterns that didn't exist before (Sağlam & Cengiz, 2022). The second drawback is the neighborhood links. The number of linkages for each sample is constant, and the number of neighbors can't vary from one sample to the next (Sağlam & Cengiz, 2022). Also, SMOTE is not recommended for high-dimensional data; otherwise, the classifier will be biased toward the minor class (Blagus & Lusa, 2013). This work employs a conditional variational autoencoder (CVAE) to handle data imbalance. Therefore, a method is required to balance the difference between classes (CVD and non-CVD).

1.2.2 Problem in Existing Machine Learning Models and Optimization Techniques in CVD Predictive Systems

It can be observed that the models mentioned in the previous works have performed well in predicting CVD (mentioned in Chapter 2- Applications of ML in detecting CVD). However, their work still has several shortcomings. Currently, models are trained using quite simple ML algorithms such as DT (Almazroi, 2022), ETC (Ishaq et al., 2021), XGB (Ahmad et al., 2022), or RFC (Ali et al., 2021). However, recent developments in ML methodologies have enabled the successful use of stack-ensemble learning framework and deep learning in computational biology and healthcare, particularly for developing

more accurate and stable models to enhance the performance of HD diagnosis. Therefore, there is room for improvement in developing a hybrid ML model for CVD detection on different CVD datasets compared to other ML classifiers' deep learning techniques. In addition, many previous works did not introduce HPO for the ML models (Ali et al., 2021; Fitriyani et al., 2020; Haq et al., 2018; Ishaq et al., 2021; Nilashi et al., 2020; Tiwari et al., 2022), where HPO automates the hyperparameter tuning process and enables users to apply ML models (Hutter et al., 2019), it increases ML models' efficiency since many ML hyperparameters have distinct optimal values for optimal performance on various datasets (Yang & Shami, 2020).

There is no literature study that has used some advanced optimization algorithm to tune the hyperparameters of hybrid models for CVD detection over multiple datasets. Furthermore, the comparison of advanced optimization algorithms with conventional optimization methods, including a genetic algorithm (GA), grid search (GS), and particle swarm optimization (PSO), is also missing. Therefore, an advanced optimization algorithm with better convergence speed is also required to tune the hyperparameters of the developed ML model to enhance its prediction accuracy in comparison with GA, RS, and PSO algorithms.

1.2.3 Problem-related to ML model interpretation

"Black-box" models are challenging to comprehend, as it is important to know why forecasts are made as well as the prediction itself. Although these strategies enhance CVD research, higher prediction accuracy can hinder model interpretability. For example, a trained RFC ML model predicts whether a specific sample is non-CVD or CVD. The model uses all the sample's attributes, such as age, gender, and chest pain, to predict whether they have a disease. Suppose the RFC model predicts a 93% chance of detection for a particular sample. How did it come to this conclusion? RFC models can easily

consist of tens or hundreds of "decision trees." This makes it nearly impossible to grasp their reasoning. However, each sample decision can be made interpretable using an approach for model interpretation. The model interpretation plots show how the model used each sample feature and reached a prediction of 93% (or 0.93). A certain conclusion can be obtained using the most important features the model factored. For instance, CVD chances increase substantially if the individual is over 40. Also, if the individual was male, the chances of disease increased even more. Finally, if the individual was not having chest pain, the chances of not having CVD disease might fall slightly. Therefore, applying these analysis algorithms in clinical fields is vital for better prediction interpretation, visualizing the most contributing features on the model's output, and showing the feature interactions.

1.3 Research Objectives

This study aims to propose a pragmatic and efficient CVD detection method that accurately classifies each individual in the clinic as a CVD or non-CVD individual. The following objectives are proposed for this research to achieve the aim of this study:

1. To develop an improved weighted random forest-based method to handle the data imbalance issue on the algorithm level.
2. To develop a conditional variational auto-encoder-based method to handle the data imbalance issue on the data level by generating new samples.
3. To build and optimize cardiovascular disease detection systems using hybrid ML model with improved prediction accuracy through proposed model architecture.
4. To investigate the proposed learning mechanisms and highlight the most contributing features enabling the proposed model to produce accurate CVD prediction outcomes through a deeper insight and model interpretation.

1.4 Research Methodology

The research methodology adopted to achieve the objectives of this research work is highlighted in this section.

1. A review of past research on the detection of CVDs is conducted to identify the most successful methodologies developed and adopted for CVD detection.
2. The problem associated with CVD prediction using an imbalanced dataset is highlighted based on the literature review.
3. Study different ML-proposed CVD detection and classification models to develop an optimal network for this research.
4. Study different balancing methods based on algorithm-level or data-level to understand their capabilities and constraints for tackling imbalanced datasets and build a balancing model to overcome the stated issues.
5. Study different optimization techniques in terms of their capabilities and limitations. The metaheuristic techniques include Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The model-free algorithms include Grid search (GS) and Random Search. The Bayesian optimization algorithms include the Gaussian Process (GP) and Tree-structured Parzen Estimator (TPE). The optimization methods will be utilized to solve the optimum hyperparameters for the balancing and classifier models.
6. Implement a model interpretation framework for a deeper insight into the proposed model working mechanisms to understand the feature interaction and impact on the proposed CVD detection system. It then highlighted the most contributing and essential features to enable the proposed model to produce improved CVD prediction outcomes.

7. To show the effectiveness of the proposed balancing and detection model, the results are verified using statistical tests and then compared with other machine or deep learning models.

1.5 Scope of Research

The CVD detection model is an interesting assistance tool for the cardiologist, which classifies each sample into its respective class, whether CVD or non-CVD. ML is the most recent and successful technique used for disease diagnosis. Several ML architectures and methodologies have been proposed and deployed to classify individuals of different classes precisely. CVD detection is a binary classification task that classifies a person into two classes, i.e., CVD and non-CVD samples. The different number of CVD and non-CVD samples in any given data introduces a class imbalanced data problem, resulting in low detection accuracy. Moreover, the outlier and missing data make it hard for the samples to be accurately predicted because they possess dual properties of both classes.

Further, it comes in the form of multiclassification for severity classification, which classifies the individual severity of CVD level. Therefore, this study critically analyses the effect of class imbalance data in the training process. It investigates the potential of ML structure and other methods on the algorithm-level and data-level to balance the natural difference between CVD and non-CVD samples. A balancing factor is generated from the ratio of CVD and non-CVD samples. Moreover, CVD can be detected using heart imaging or Electrocardiography (ECG) to predict whether an individual has CVD. However, this study focuses on the detection of CVD through clinical data.

In addition, the hyperparameters of the developed hybrid ML model are optimized further to improve the prediction accuracy of CVD, using TPE-BO optimization compared with different optimization methods such as PSO, GA, and RS. Various datasets that depict complicated and real-world problems are utilized to demonstrate the

accuracy of the proposed methods. Cleveland, Statlog, Z-Alizadeh, and heart disease clinical records are the only datasets used to validate and evaluate the performance of the proposed methods. Using Python frameworks like scikit-learn and TensorFlow, several CVD detection models are developed and trained. Python is also used for any dataset preprocessing, such as normalization and scaling. Based on its classification accuracy and misclassification error rate, the performance of an ML model utilizing a particular dataset is evaluated.

In summary, the proposed CVD detection methods have great potential not only in the CVD detection field but also in other fields such as breast cancer, diabetes, and so on.

1.6 Research Outline

This thesis consists of five chapters. **Chapter 1** presents the overview of the CVD detection system, problem statement, research objective, and scope of the research.

Chapter 2 presents a concise and comprehensive literature review on machine learning, deep learning, balancing methods, network architectures, CVD detection systems, and existing research.

Chapter 3 describes the methodology of the proposed solution, including model architecture, balancing method, system configuration, and hardware description.

Chapter 4 evaluates the experimental results obtained from the simulations and the performance of the proposed method. The proposed method's performance is further analyzed compared to existing research.

Chapter 5 summarizes the research work, the current limitation of the proposed work, and proposes directions for further improvement in the future.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

Within this chapter, an exhaustive exposition of cardiovascular diseases is provided with the aim of elucidating the condition and distinguishing it from other diseases. Furthermore, a comprehensive delineation of data preprocessing is presented, encompassing data cleansing, transformation, rebalancing, and the assessment of feature significance. Additionally, an in-depth discussion on the utilization of machine learning methodologies for addressing this problem is conducted. Lastly, a critical examination of the application of machine learning in the context of heart disease detection is undertaken, drawing insights from existing literature and culminating in a summarized conclusion for this chapter.

2.2 An overview of heart disease

Cardiovascular disease (CVD) encompasses a spectrum of pathological conditions, including but not limited to hypertension, stroke, heart failure (HF), and arrhythmia (Cheriyan et al., 2010). It is a leading global cause of mortality, with a projected escalation in worldwide fatalities to approximately 23.6 million by the year 2030 (Zaragoza et al., 2011). Traditional clinical diagnostic modalities, such as chest x-rays, echocardiograms, and electrocardiograms, have conventionally served as tools for the diagnosis and ongoing monitoring of cardiovascular health (Bonnefont-Rousselot, 2016). The assessment of cardiovascular risk often relies on established factors such as cholesterol levels, diabetes status, age, smoking habits, and hypertension. However, the reliability of these factors in routine clinical evaluation is not infallible. This can result in individuals without a high predisposition to CVD receiving unnecessary preventative measures. Consequently, there is a pressing need for more refined methodologies to distinguish between individuals with CVD and those without. One promising avenue is the utilization

of machine learning techniques for the enhanced identification of heart disease (Weng et al., 2017).

2.3 Data Preprocessing

The preparatory procedures conducted on data prior to its utilization in an algorithm are commonly referred to as data pre-processing. Data pre-processing is the systematic conversion of unrefined data into a refined dataset. Essentially, when data is amassed from diverse sources, it often lacks the structure required for meaningful analysis. Therefore, data preprocessing, an integral facet of the data analysis and knowledge extraction process within a database framework, assumes paramount importance. In scenarios characterized by copious amounts of irrelevant, redundant, noisy, or unreliable data, the process of knowledge extraction during analysis and mining phases becomes substantially more arduous. It is during this phase that raw data undergoes a series of transformations to render it comprehensible and amenable to analysis (Saboor et al., 2022). Typical stages in data preprocessing encompass data cleansing, transformation, feature selection, and data balancing.

2.3.1 Data cleaning and transformation

Data cleansing constitutes the systematic procedure of identifying and subsequently eliminating deceptive, erroneous, or irrelevant data entries, often involving corrective actions such as data replacement, modification, or removal. This process is further facilitated through operations such as imputing missing values, smoothing noisy data, and rectifying inconsistencies within the dataset (Bhatt et al., 2023).

Conversely, data transformation embodies the act of converting data from one format to another, typically transitioning from the format native to a source system to one compatible with the requirements of a destination system. Data transformation serves as

an integral component of numerous data integration and data management endeavors, including data wrangling and data warehousing.

Within the domain of data transformation, several methods for data normalization exist, including the employment of techniques such as MinMaxScaler and standard deviation. MinMaxScaler normalization, for instance, is particularly advantageous when machine learning algorithms perform optimally with features of comparable scales and data distributions that closely approximate normality. In essence, scaling entails the alteration of value ranges without modifying the underlying distribution's shape. Typically, this operation results in a range between 0 and 1. MinMaxScaler achieves this by subtracting the minimum feature value from each data point and subsequently dividing by the range, where the range represents the difference between the highest and lowest original values. Importantly, MinMaxScaler maintains the original distribution's shape and preserves the information content contained within the initial dataset, without significantly diminishing the influence of outliers. On the other hand, StandardScaler normalizes a feature by firstly subtracting the mean and then scaling to achieve unit variance. Unit variance is attained by dividing each data point by the standard deviation. It is worth noting that StandardScaler deviates from the stringent definition of scaling outlined previously. In the distribution produced by StandardScaler, both the standard deviation and variance assume a value of 1. Therefore, the distribution is characterized by a mean of 0, with approximately 68% of data points falling within the range of -1 to 1 (Ramesh et al., 2022).

2.3.2 Feature importance

The concept of feature significance serves as a critical determinant of the extent to which individual features contribute to a model's predictive capacity. It quantifies the degree of usefulness associated with a specific variable within the context of the current

model and its predictive capabilities. In a study conducted by Kursa and Rudnicki (Kursa & Rudnicki, 2011), it was observed that Random Forests are commonly leveraged within data science workflows for the purpose of feature selection. This preference is underpinned by the inherent characteristics of tree-based techniques employed by Random Forests, which naturally rank features based on their ability to enhance node purity. This enhancement entails a reduction in impurity levels across all constituent trees. It is noteworthy that nodes exhibiting the most substantial reduction in impurity are typically encountered at the initial stages of tree development, while nodes with minimal impurity reduction tend to manifest towards the latter phases of tree growth. Consequently, by strategically pruning trees below a designated node, it becomes feasible to derive a subset of the most indispensable features for subsequent model construction and analysis (Pathan et al., 2022).

2.4 Data balancing

A balanced dataset is characterized by a relatively equitable distribution of labels, with labels denoting the categorical assignments associated with individual data points. To illustrate, consider a dataset encompassing two distinct classes, such as 'male' and 'female.' In the context of a balanced dataset, the distribution approximately allocates an equal share to each class, resulting in a nearly 50% representation for both males and females. On the contrary, an unbalanced dataset manifests when there exists a significant discrepancy in class memberships. Employing the male and female classes as an exemplar, an unbalanced dataset may exhibit a substantial imbalance between the two groups, with one class significantly outnumbering the other. In light of the implications stemming from imbalanced datasets, it is reassuring to note the existence of viable solutions to rectify such disparities. In the ensuing discussion, we will explore several of these solutions in detail (Nagavelli et al., 2022).

2.4.1 Random Over-sampling

Random oversampling, as elucidated by Chawla, Bowyer, Hall, and Kegelmeyer (Chawla et al., 2002), entails the augmentation of training data by introducing additional instances of select minority classes. The oversampling process may be iterated multiple times, and rather than merely replicating each sample from the minority class, a technique involving random selection with replacement may be employed, thus enhancing the diversity of the augmented dataset. Random oversampling represents a non-heuristic approach, which aims to rectify class distribution imbalances through the stochastic duplication of minority class instances. This technique, despite its simplicity, demonstrates a high level of competitiveness when juxtaposed with more intricate oversampling methodologies. Furthermore, it is computationally economical in comparison to alternative methods that yield substantial performance enhancements (Batista et al., 2004). To illustrate the application of random oversampling, let us consider a binary classification problem encompassing two classes and a dataset comprising one hundred thousand data points. In this scenario, the positive class, denoting the minority class, comprises 20,000 instances, while the negative class contains 80,000 instances. To achieve class balance, the positive class is oversampled by replicating its 20,000 data points fourfold, resulting in a total of 80,000 instances for both the positive and negative classes. Accordingly, the dataset's size is expanded to 160,000 instances) Chaudhuri et al., 2024((Chaudhuri et al., 2024).

In cases involving undersampling, particularly when dealing with a highly prevalent class, the primary objective is to diminish the representation of the majority class to achieve dataset balance. To illustrate this concept, consider a binary classification problem featuring two classes and a dataset comprising one hundred thousand data points. In this scenario, the positive class consists of 20,000 instances, while the negative class encompasses 80,000 instances. The task at hand is to perform undersampling on the

majority class. This involves the random selection of 20,000 data points from the pool of 80,000 available instances. As a result, we obtain 20,000 positive data points and 20,000 negative data points, resulting in a dataset totaling 40,000 instances. Tomek linkages represent a technique utilized to address classification challenges and enhance data classification precision by minimizing class label noise. The primary objective of this technique is to eliminate as much class label noise as possible, whereby label noise refers to alterations in the assigned labels associated with instances. In the context of classification, each instance is associated with a label that signifies its category. Label noise may occur due to various factors, resulting in inaccurate labels. Class noise, a subset of label noise, pertains specifically to instances where observable labels have been modified inappropriately, such as erroneously assigning a positive label to a negative instance. Tomek linkages serve to identify instances that are borderline, carrying a higher risk of misclassification. Subsequently, these instances are subject to removal, a process known as Tomek link deletion. Tomek linkages pertain to points characterized by distinct class labels that serve as nearest neighbors to one another. This technique facilitates the identification and elimination of instances near different class labels, effectively eradicating undesired class overlap. Consequently, only instances with neighbors belonging to the same class are retained, thereby reducing class imbalance (Albert et al., 2023).

Comparatively, the performance of a Support Vector Machine (SVM) algorithm bears similarity to that of the Tomek links approach. SVMs are adept at classification tasks, as they establish a hyperplane decision boundary that effectively separates samples into distinct groups, making them suitable for both classification and regression tasks. This hyperplane is characterized by a margin that maximizes the distance between the boundary and the nearest instances of each class. However, SVMs exhibit sensitivity to imbalanced datasets and may yield suboptimal outcomes. Nevertheless, SVM

performance on imbalanced data can be enhanced by adjusting a parameter denoted as C, which governs the trade-off between expanding the margin between classes and minimizing misclassification instances. In the context of imbalanced datasets, the C value can be weighted to reflect the relative importance of each class, thereby enabling SVMs to operate effectively with such data. This variant of SVM, referred to as Weighted SVM or Cost-Sensitive SVM, accommodates the intricacies of imbalanced datasets.

2.4.2 Synthetic Minority Over-sampling Technique (SMOTE).

The Synthetic Minority Over-sampling Technique (SMOTE) represents an advanced approach to oversampling, designed to enhance the effectiveness of random oversampling. It accomplishes this by generating new synthetic instances that lie along the linear path between minority class examples and their specifically chosen nearest neighbors (Blagus & Lusa, 2013). SMOTE operates by creating novel minority instances through the amalgamation of existing minority instances, effectively constructing virtual training records for the minority class via linear interpolation. These synthetic training records are generated by randomly selecting one or more of the k-nearest neighbors for each instance within the minority class (Hussain et al., 2022).

Following the completion of the oversampling process, the dataset undergoes a reconstruction phase, and subsequently, various classification models can be applied to the processed data. The application of SMOTE renders decision regions less constrained and more expansive. However, it's important to note that while balance is achieved, no new or additional information is introduced to the model. SMOTE serves as a means of oversampling that differs from traditional replication of minority class instances. Instead, SMOTE generates entirely new instances by selecting those in close proximity within the feature space. This process involves identifying the k-nearest neighbors (k-NN) within

the minority class. The k-nearest neighbors method involves classifying data points based on their proximity to other data points within the dataset.

SMOTE proceeds by randomly selecting an instance from the minority class and computing its k-NN. Subsequently, one of the neighboring instances is chosen at random, and a synthetic example is created at a randomly determined location along the segment connecting the two instances. This process can be repeated as needed to generate the requisite number of synthetic instances for the minority class to achieve balance. An advantage of oversampling techniques like SMOTE is the absence of data loss from the original training set, as both majority and minority class data are utilized in their entirety. However, it is important to be aware of the potential drawback of oversampling, which is the risk of overfitting the model (Muntasir Nishat et al., 2022).

2.5 Machine learning algorithms

Presently, artificial intelligence (AI) stands as a transformative force impacting multiple industries, such as banking and medical diagnosis, demonstrating its prowess in tackling intricate problems (Marwala & Xing, 2018). The advent of machine learning (ML) has notably expedited the progress of artificial intelligence (Cioffi et al., 2020). Machine learning, encompassing both the academic discipline and its practical techniques, represents a subset of AI. In recent years, ML has emerged as the linchpin for advancing AI, finding widespread application in both industry and academia to create predictive models capable of delivering accurate outcomes in highly complex scenarios (Sidey-Gibbons & Sidey-Gibbons, 2019). Many achievements in machine learning hold the potential for further exploration and enhancement, particularly in domains characterized by imbalanced datasets such as credit risk prediction and medical diagnosis. This chapter offers a comprehensive overview of numerous machine learning applications within these domains. Furthermore, it introduces the two primary categories of machine

learning, namely supervised and unsupervised machine learning. Additionally, this chapter furnishes a mathematical exposition of the algorithms utilized throughout the dissertation, exemplifying the supervised and unsupervised machine learning models, as illustrated in Figure 2.1 (Ramesh et al., 2022).

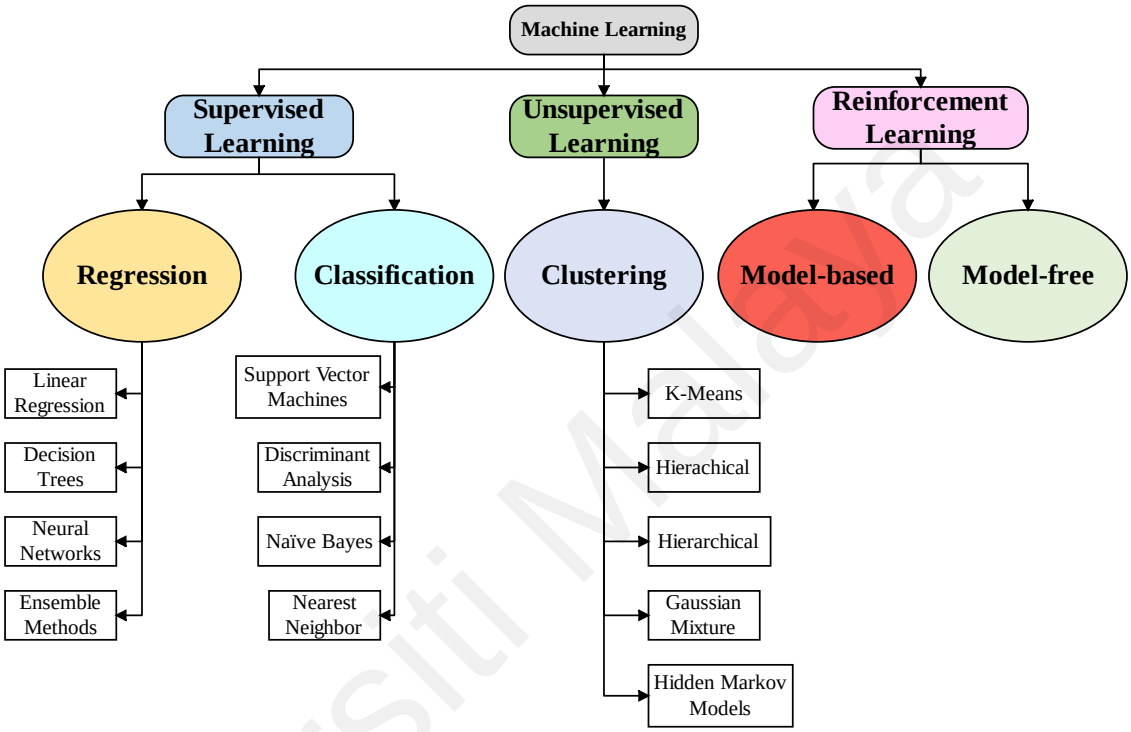


Figure 2.1: Machine learning classification models.

2.5.1 Supervised learning

Supervised learning stands out as the predominant paradigm in the realm of machine learning (Bengio & LeCun, 2007). It is characterized by the utilization of datasets featuring known target variables to train models. In the context of supervised learning, these target variables may assume either discrete or continuous forms. In instances where the target variable takes on a discrete value, the process is recognized as classification. For instance, it encompasses tasks such as predicting the creditworthiness of an applicant (creditworthy or not), categorizing clients as good or bad, or determining the presence or absence of a disease (Sidey-Gibbons & Sidey-Gibbons, 2019).

Classification in supervised learning encompasses a spectrum of algorithms, including logistic regression, naïve Bayes, random forest, support vector machines, and neural networks. Conversely, when the target variable assumes continuous values, supervised learning is termed regression. Regression methodologies enable predictions based on continuous response variables, leveraging the insights gleaned during the training phase. Various regression algorithms are available, including linear regression, multivariate regression, and lasso regression. The selection of a particular regression analysis method is contingent upon factors such as data attributes, response variables, and the inherent characteristics of the regression curve. This regression curve serves as a visual representation of the relationship between predictor and predicted variables (C. Gupta et al., 2022).

2.5.1.1 Decision tree algorithm

Decision tree algorithms are a class of supervised machine learning methods used for both classification and regression tasks (Topîrceanu & Grosseck, 2017). In the context of classification, decision trees are specifically referred to as classification trees, where the predicted variable comprises a binary set of values. Conversely, when the predicted variable takes on continuous values, it is termed a regression tree. Decision trees, known for their simplicity, find widespread utility across various applications (Ozcan & Peker, 2023).

A decision tree is comprised of three fundamental components: the root node, leaf nodes, and branches. The tree construction process commences with the root node, which, along with the leaf nodes, contains questions or criteria that must be satisfied. The branches, depicted as arrows connecting nodes, signify the flow from questions to corresponding answers. Several tree-based machine learning algorithms exist, including Classification and Regression Tree (CART), Iterative Dichotomiser 3 (ID3), and C4.5.

To calculate the Gini index for a data sample with classes, one can employ the following assumptions, as indicated in Eq 2.1:

$$Gini = \sum_{i=1}^{J} p_i \sum_{k=1}^{J} p_k = \sum_{i=1}^{J} p_i\,(1 - p_i) = \sum_{i=1}^{J} (p_i - {p_i}^2) = \sum_{i=1}^{J} p_i - \sum_{i=1}^{J} {p_i}^2 = 1 - \sum_{i=1}^{J} {p_i}^2 \quad 2.1$$

where denotes the probability that an instance is classified into a specific class (Loh, 2011).

2.5.1.2 Support vector machine

Support Vector Machine (SVM) is a machine learning algorithm suitable for addressing both regression and classification problems. Grounded in the principles of statistical learning, SVM has demonstrated its proficiency in delivering accurate predictions across diverse domains (Marwala, 2014). SVM's versatility extends to linear classification tasks, and it proves invaluable in resolving non-linear classification challenges through the application of the kernel method (Kafai & Eshghi, 2017). The kernel method effectively transforms non-linearly separable input data into a higher-dimensional space, within which a hyperplane capable of effectively segregating the data is constructed. SVM offers a selection of kernels, encompassing polynomial, radial basis, linear, Gaussian, and various non-linear kernels (Faieq & Mijwil, 2022).

Considering the input data as T$\{(x_i, y_i)_N\}$, where $y_i \in \{+1, -1\}$, the primary objective of the SVM classifier is to determine a hyperplane that effectively partitions the feature space into two distinct regions corresponding to the classes within the input data (Kafai & Eshghi, 2017). In this context, a hyperplane denotes a linear function of *x*, denoted as $f(x) = \langle w.b \rangle + b$, as demonstrated in Eqs 2.2 and 2.3.

$$y_i\big(f(x)\big) = y_i(\langle \mathrm{w.b}\rangle + \mathrm{b}) > 0 \quad (2.2)$$

$$\big(f(x)\big) = (\langle \mathrm{w.b}\rangle + \mathrm{b}) = 0 \quad (2.3)$$

where *w* denotes a weight vector, *b* represents bias, whose value is a scalar quantity.

The remarkable generalization capability of the SVM stems from its inherent ability to minimize the generalization error while simultaneously maximizing the separation margin. This optimization problem is formally expressed and addressed through constrained optimization techniques, specifically, the minimization of $\frac{1}{2}\|\mathrm{w}\|^2$ or equivalently, the maximization of the margin $\frac{2}{\|\mathrm{w}\|}$ with respect to $y_i(\langle \mathrm{w.b}\rangle + \mathrm{b}) > 1$. The application of the Lagrange multipliers strategy is instrumental in solving this constrained optimization problem. Upon computing the Lagrange function (denoted as *L*) and introducing an undetermined scalar α, the following relationship is derived as shown in Eqs (2.4 and 2.5):

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \quad (2.4)$$

$$L = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j x_i x_j \quad (2.5)$$

The Lagrangian function is optimized to determine the coefficients α_i while adhering to the given constraint, as shown in Eq 2.6:

$$\sum_{i=1}^{N} \alpha_i y_i = 0, \alpha_i > 0 \quad (2.6)$$

Once the coefficients α_i are obtained, a hypothesis is derived, which corresponds to a linear combination of the input data points. Subsequently, the decision function is formulated as in Eq 2.7.

$$h(x) = sng(\langle w.x \rangle + b) = sng(\langle \sum_{i=1}^{N} \alpha_j y_j x_j.x \rangle + b) \quad 2.7$$

Equation (2.7) reveals that SVM learning relies on the inner products of input pairs, while the prediction of unseen samples is entirely contingent on the inner product between the sample under consideration and the input or training data. Moreover, SVM is well-suited for scenarios with small datasets, and its performance tends to degrade when dealing with larger datasets.

2.5.1.3 k-Nearest Neighbors

The K-nearest neighbors (KNN) algorithm is a versatile machine learning technique capable of performing both classification and regression tasks. Nevertheless, its primary utilization is in classification, and it is characterized as a lazy learning and non-parametric algorithm (Lestari & Sumarlinda, 2022). It earns the non-parametric label because it refrains from making any underlying assumptions about the input data. Furthermore, KNN is classified as lazy learning due to its characteristic of adapting to data patterns upon query (Lestari & Sumarlinda, 2022). In practical terms, KNN classifies unlabeled samples by assigning them to the class of labeled samples with which they share the highest similarity. Several distance metrics can be employed for KNN computations, including Hamming, Manhattan, and Euclidean distance. For most applications, Euclidean distance is a prevalent choice (Raj & Thinakaran, 2022), and it can be expressed mathematically as:

$$D(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} \quad 2.8$$

In Equation (2.8), p and q denote the samples under comparison, each possessing distinct features. In the context of applying the KNN algorithm, a crucial parameter, denoted as 'k,' must be specified. This parameter signifies the quantity of nearest data points, often referred to as neighbors, considered during the algorithm's execution. The

KNN algorithm is characterized by its simplicity in implementation and has found widespread application across various domains, including but not limited to credit risk prediction and medical diagnosis.

2.5.1.4 Naïve Bayes Classifier

The Naïve Bayes classifier is a machine learning algorithm that is rooted in Bayes' theorem. It bears the name "naïve" because it makes the simplifying assumption that the input features are mutually independent (Chen et al., 2020). Various forms of naïve Bayes classifiers, including multinomial and Gaussian naïve Bayes, have been developed and are predominantly applied in scenarios involving extensive datasets (Huang & Li, 2011). In accordance with Bayes' theorem, the class variable (c) for a given sample data point (x) is computed by evaluating the posterior probability, denoted as P(c|x), as follows:

$$D(c|x) = \frac{P(x|c)P(c)}{P(x)} \tag{2.9}$$

In equation 2.9, *(c|x)* signifies the posterior probability of sample data x given class c, while P(c) represents the prior probability of the class variable *c*. *P(x)* corresponds to the prior probability of the sample data x.

2.5.1.5 Logistic regression

Logistic regression is a statistical modeling approach employed for the analysis of datasets containing multiple predictor variables to predict a binary response variable (Bejjanki et al., 2020). Logistic regression is particularly advantageous in scenarios where the class attributes exhibit binary characteristics, making it a valuable tool in credit risk prediction and medical diagnostics. Furthermore, this method aims to construct a model that best characterizes the relationship between the response variable and predictor variables, yielding the following formulated variables, as demonstrated by Eq (2.10).

$$logit(p) = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \cdots + b_kX_k \quad 2.10$$

In this context, " p " represents the probability of the presence of the attribute under consideration. This entails a logit transformation of the likelihood of the attribute's presence. Additionally, the logit transformation is visualized as the natural logarithm of the odds, as depicted by Eqs 2.11-2.12.

$$odds = \frac{p}{1-p} = \frac{probability\ of\ presence\ of\ characteristic}{probability\ of\ absence\ of\ characteristic} \quad 2.11$$

$$logit(p) = \ln\left(\frac{p}{1-p}\right) \quad 2.12$$

Another iteration of logistic regression is the softmax regression, alternatively known as multinomial logistic regression. It is employed to construct models designed for datasets containing multiple class variables. The softmax function is mathematically defined as represented in Eq 2.13.

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{x_i}} (i = 1, 2, \ldots., N) \quad 2.13$$

where $x_1, x_2, , , x_N$ represent the input values, and $f(x_i)$ is the output, representing the probability of the sample belonging to the *i*-th class. Throughout this work, logistic regression is employed.

2.5.1.6 Random Forest

Random Forest Regression (RFR) is a tree-based regression technique developed by Breiman, which involves the construction of a substantial number of regression trees (Breiman, 2001). RFR has garnered considerable attention in recent years due to its exceptional performance, ease of implementation, and computational efficiency. Essentially, a random forest consists of an ensemble of tree predictors, denoted as $\mathrm{fn}(\mathrm{X}; \theta n)$. Each tree is a collection of if-statements that can be visualized in a tree structure, akin to graph theory. The task of identifying an optimal set of if-statements that

aligns with the observed data is referred to as model development or training. Therefore, training entails an optimization process guided by an objective function aimed at minimizing the discrepancy between the predicted value $\mathrm{fn}(\mathrm{x}_i; \theta n)$ and the observed values y_i, as expressed in Eq. (2.14).

$$\text{Objective function} = \min_{\theta} \sum_i (\mathrm{fn}(\mathrm{x}_i; \theta n) - \mathrm{y}_i)^2 \qquad 2.14$$

The function $\mathrm{fn}(\mathrm{x}_i; \theta n)$ is typically chosen to represent the mean of the observations that satisfy a particular if-statement condition. A pertinent concern may arise regarding the prevention of the model from generating overly extensive if-statements tailored to each record in the dataset. To mitigate such issues, constraints are imposed on the tree in terms of maximum depth and a predefined limit on the number of branches. These constraints act as safeguards against overfitting. Additionally, the model's performance is assessed using a dataset that was not part of the training data, ensuring its ability to generalize. For a detailed exploration of the convergence equations of random forests, please refer to the provided source. The ultimate prediction produced by a random forest regressor, denoted as $Ypred$, is the averaged estimate derived from N individual trees, as illustrated in Eq (2.15).

$$Ypred = \frac{\sum_{n=1}^{N} \mathrm{fn}(\mathrm{X}; \theta n)}{N} \qquad 2.15$$

2.5.2 Unsupervised learning

Unsupervised learning, a machine learning paradigm, operates on data without the guidance of explicit class labels, aiming to extract meaningful insights and patterns (Aïmeur et al., 2013). One prominent application of unsupervised learning is clustering, a technique commonly employed in exploratory data analysis to unveil latent structures within data. Principal component analysis and autoencoders are additional techniques that fall within this category. Furthermore, there exists a specialized branch of machine

learning termed semi-supervised learning, wherein algorithms leverage a combination of labeled and unlabeled data for training. Semi-supervised learning represents the intersection of supervised and unsupervised learning methodologies (Van Engelen & Hoos, 2020).

2.5.2.1 Clustering

In the realm of unsupervised ML, a cluster is defined as a grouping of data points that exhibit similarity within the group while demonstrating dissimilarity with other clusters (Krittanawong et al., 2017). Various clustering algorithms, such as k-means, k-harmonic means, and hierarchical clustering algorithms, can discern different clusters within unlabeled data. Subsequently, the clustered or grouped data can be subjected to further analysis (Luo et al., 2011). It is crucial to note that the primary objective of clustering is not to categorize, approximate, or predict specific data values. Instead, it focuses on partitioning the data into homogeneous clusters of records, where the variance among different clusters is significantly higher than the variance within each individual cluster (Maimon & Rokach, 2005). Among the most renowned clustering algorithms, k-means clustering stands out. This algorithm establishes a user-defined number of centroids to delineate data clusters. Each data point is assigned to its nearest centroid, and through iterative calculations on the centroids, they are optimized until the desired number of predefined iterations is achieved (Kilic, 2020). Hierarchical clustering, another prominent algorithm, organizes data into a hierarchy of clusters, creating a structure with a single encompassing cluster at the highest level and individual object singleton clusters at the lowest level. This process culminates in the construction of a dendrogram (Murtagh & Contreras, 2012).

2.5.2.2 Principal component analysis

Principal component analysis (PCA) is a fundamental algorithm that facilitates the transformation of the initial variables within a dataset into a novel set of orthogonal variables recognized as principal components (Abdi & Williams, 2010). This popular unsupervised ML technique, PCA, is extensively applied for feature extraction. It accomplishes this by projecting the original parameter vectors into a fresh feature space through the utilization of a linear transformation matrix (Wang & Paliwal, 2003). Additionally, PCA is adept at reducing the dimensionality of a dataset, thereby converting high-dimensional data into a lower-dimensional format while endeavoring to retain the maximum degree of variability inherent in the dataset (Nie et al., 2014). One of the underlying assumptions of PCA is grounded in the belief that the bulk of information contained within the dataset is concentrated in the directions characterized by the most substantial variations (Wang & Paliwal, 2003).

2.6 Deep Learning

Deep learning (DL), an increasingly prominent subfield within the realm of artificial intelligence (AI), has made significant strides in tackling a wide array of intricate challenges. DL has achieved remarkable breakthroughs in diverse domains, encompassing image recognition, classification, and segmentation (Krizhevsky et al., 2017), speech recognition (Sainath et al., 2015), genomics (Alipanahi et al., 2015), reconstruction of brain circuits (Helmstaedter et al., 2013), natural language understanding (Collobert et al., 2011), and recognition of heart sounds (Chen et al., 2016). DL, in essence, emulates the intricate processing patterns of the human brain and typically consists of multi-layered artificial neural networks. It leverages data to scrutinize and comprehend intricate hierarchical representations featuring multiple levels of abstraction (LeCun et al., 2015). The widespread application of compute-intensive DL methodologies has been made feasible by the technical progress in graphics processing

units and the advent of cloud computing, thus ushering in transformative capabilities for DL (Raina et al., 2009). Among the most prevalent DL-based neural network algorithms are the Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN). As an illustrative instance, Xiong et al. devised a DL-based model known as RhythmNet, which seamlessly integrates both CNN and RNN approaches. This innovative fusion, known as RhythmNet, was tailored for the classification and diagnosis of atrial fibrillation (AF) utilizing ECG data.

2.6.1 Convolutional Neural Network

CNN is an artificial neural network structure characterized by multiple layers, including a convolutional layer designed to extract hierarchical features from raw input data. It is further complemented by fully connected layers that serve as classifiers (Xia et al., 2018). Notably, CNN has garnered extensive utilization in the domains of image processing and classification (Gulshan et al., 2016). Furthermore, CNN has found wide-ranging applications in the realm of arrhythmia detection, particularly through the utilization of publicly available ECG databases. These applications have yielded impressive results, achieving an accuracy exceeding 75% (Hannun et al., 2019). The work conducted by Hannun et al. is particularly noteworthy, as they harnessed CNN to classify various arrhythmia conditions. They accomplished this by analyzing single-lead ECG data derived from a substantial patient population, ultimately attaining an impressive Area Under the Curve (AUC) value of 0.97 (Hannun et al., 2019). This research underscores the promising potential of DL within clinical contexts, showcasing its capacity to significantly enhance the diagnosis and categorization of arrhythmias.

2.6.2 Recurrent Neural Network

RNN, or Recurrent Neural Network, represents a potent dynamic system tailored to model neural sequences, particularly focusing on the identification of temporal patterns

within longitudinal data (Choi et al., 2017). It has proven instrumental in tackling intricate machine learning tasks, including tasks related to linguistic phrase acquisition and the generation of coherent natural language descriptions for images and their constituent regions (Choi et al., 2017). In a noteworthy application, Choi et al. harnessed the capabilities of RNN to explore longitudinal electronic health record data, with a particular focus on discerning relationships among time-stamped events, such as disease diagnoses and medication orders (Choi et al., 2017).This research serves as a compelling demonstration of RNN's potential, notably in the context of incident heart failure detection. The study achieved an AUC of 0.777 when using data collected over a 12-month observation window, and an even more impressive AUC of 0.883 when extending the observation window to 18 months (Choi et al., 2017). These results highlight the promising prospects of RNN in the realm of healthcare applications.

2.7 Application on heart disease detection using machine learning.

In the subsequent sections, we will delve into a comprehensive discussion of the utilization of various machine learning models for the purpose of detecting heart disease.

2.7.1 Applications of Logistic Regression

Over the years, logistic regression has gained extensive utility across various predictive tasks, including its application in medical diagnosis prediction (Tortajada et al., 2015). Recently, an approach was introduced for predicting diabetes (Zhu et al., 2019). This method incorporated PCA to enhance the predictive capabilities of both KNN and logistic regression. PCA, a mathematical algorithm, aims to reduce the dimensionality of input data while preserving the inherent variability within the dataset. This reduction is achieved by identifying the principal components, which are directions in the data space along which variability is maximized. The application of PCA to the data led to a 1.98% improvement in the accuracy of the logistic regression classifier.

Furthermore, logistic regression has found utility in predicting sepsis-related mortality (Ribas et al., 2012). The proposed approach involved the analysis of sepsis indicators through feature extraction employing a latent model. Simulation results demonstrated a notable enhancement in classifier performance. Sepsis, characterized by an exaggerated response to bacterial infections in the bloodstream, stands as a leading cause of mortality among Intensive Care Unit (ICU) patients (Keeley et al., 2017), (Thompson et al., 2019). Thus, research focused on predicting sepsis-related mortality holds considerable significance.

2.7.2 Applications of Support vector machine

In the realm of medical diagnosis, SVM have been harnessed for the prediction of a wide array of diseases, including instances such as diabetes and breast cancer detection (Gürbüz & Kılıç, 2014). This approach introduced a feature adaptivity mechanism aimed at expediting computational processes while concurrently augmenting predictive accuracy. The proposed algorithm demonstrated superior performance in contrast to conventional SVM methodologies. Referred to as "adaptive SVM," this algorithm exhibited outstanding predictive capabilities in the context of diabetes and breast cancer, achieving a remarkable accuracy rate of 100% in both instances. Furthermore, an empirical evaluation was undertaken to compare the performance of various Machine Learning (ML) algorithms using a heart disease dataset (Hussain et al., 2020). The ensemble of algorithms encompassed several SVM kernels in addition to other ML techniques, including decision trees, KNN, and an ensemble classifier. The SVM kernels examined in the study encompassed Gaussian, linear, radial basis function, and polynomial kernels. The experimental outcomes distinctly favored the linear kernel, which yielded superior performance metrics, including an AUC value of 0.97 and an accuracy rate of 93.1%.

2.7.3 Applications of k-Nearest Neighbors

KNN has found extensive application in the domain of disease prediction (Dalen et al., 2014; Qin et al., 2014). One notable approach entailed the prediction of heart disease through the synergistic integration of a genetic algorithm and KNN (Deekshatulu & Chandra, 2013). This methodology revolved around the prioritization of attributes based on their significance and the elimination of extraneous features employing genetic search as a metric of utility. Notably, by training KNN on the most pivotal attributes, a discernible enhancement in predictive performance was observed. In a separate study conducted by Sowmiya and Sumitra (Sowmiya & Sumitra, 2021), an innovative technique was introduced for heart disease prediction, leveraging the ant colony optimization method for feature selection. A hybrid KNN classifier was subsequently employed for the predictive task. This approach yielded a classification accuracy of 99.2%, manifesting a remarkable level of performance superiority when contrasted with various other machine learning classifiers, including decision trees, naïve Bayes, SVM, and traditional KNN. Furthermore, another research endeavor harnessed KNN to anticipate the two-year risk of type 2 diabetes mellitus development in individuals with prediabetes (Garcia-Carretero et al., 2020). The dataset utilized for algorithm training encompassed 1647 samples, featuring clinical and laboratory test-derived attributes. Remarkably, the KNN classifier attained a test accuracy rate of 96%, coupled with a true negative rate of 78% and a true positive rate of 99%.

2.7.4 Applications of naïve Bayes

A NB classifiers have found substantial utility in the realm of medical diagnosis. A Gaussian NB model was harnessed for the prediction of lung and breast cancers in a study conducted by Kamel, Abdulah, and Al-Tuwaijari (Kamel et al., 2019), attaining test accuracies of 90% and 98%, respectively. In a separate endeavor, naïve Bayes was enlisted to classify melanoma (skin cancer) as either malignant or benign using

epiluminescence microscopy-derived images as input data (Arasi et al., 2018). Comparative analysis with a decision tree indicated the superiority of the NB classifier, boasting an accuracy rate of 98.8%, while the latter achieved an accuracy of 92.86%.

Moreover, a novel Hidden Naïve Bayes (HNB) approach was proposed to discern heart disease by Jabbar and Samreen (Jabbar & Samreen, 2016). Diverging from the traditional naïve Bayes, HNB modifies the independence assumptions among predictor variables. Impressively, the HNB classifier achieved a test accuracy of 100%. NB-based models have been proficiently employed in predicting and detecting various heart diseases (Quesada et al., 2019). Vembandasamy et al. (2015) utilized clinical data from approximately 500 diabetic patients (exact numbers not specified in the article), encompassing attributes like age, gender, serum cholesterol, resting blood pressure, fasting blood sugar, and chest pain type, to train an NB model for predictive diagnostics of heart diseases. This model successfully discriminated against individuals with or without heart diseases, achieving an accuracy of approximately 86.4% (Vembandasamy et al., 2015).

In another investigation, an NB classifier was applied to analyze 303 observations from the Cleveland Clinic Foundation, focusing on 14 clinical parameters, including age, gender, cholesterol levels, exercise-induced angina, resting electrocardiographic results, and resting blood pressure, to diagnose heart diseases. The trained NB classifier effectively categorized patients into different risk levels of heart disease, with values ranging from "0", indicating no risk, to "4", signifying the highest risk (Medhekar et al., 2013).

Additionally, in a cross-sectional study involving 1187 participants, the NB model demonstrated its capacity to predict the necessity for coronary angiography, employing features such as gender, age, and fasting blood glucose for training. This research yielded

an AUC of 0.74 and a sensitivity of 0.892 (Golpour et al., 2020). Dekamin et al. delved into the data of 303 individuals, incorporating 54 features collected from a Tehran-based hospital, and utilized NB as one of the algorithms for diagnosing coronary artery disease, achieving an accuracy rate of 86.36% (Dekamin & Sheibatolhamdi, 2017).

2.7.5 Applications of Decision Tree

The DT algorithm, one of the pioneering supervised machine learning methodologies, derives its name from its inherent capacity to facilitate decision-making processes (Sitar-tăut et al., 2009). This algorithm constructs a tree-like model within the supervised learning framework, where nodes represent features, and branches denote the outcomes of tests associated with their respective nodes. Decision trees are useful in solving classification problems (Aljaaf et al., 2015). When traversing the tree from its root to the leaves for sample classification, a thorough examination of each node along the path culminates in assigning each sample to its predicted class (Uddin et al., 2019). Decision tree-based machine learning classifiers have played a pivotal role in various studies concerning predictive diagnostics of heart diseases (Aljaaf et al., 2015). For instance, in a study leveraging the heart disease dataset from the Cleveland Clinic Foundation, encompassing 297 patients with heart disease, a decision tree model was trained to stratify patients into five risk categories corresponding to different stages of heart failure. This endeavor yielded an average AUC of 0.91, a sensitivity of 0.865, and a specificity of 0.955 (Aljaaf et al., 2015). Puyalnithi et al. devised a risk assessment model based on decision trees, employing medical and behavioral datasets obtained from general screening processes, resulting in an AUC of 0.94 and a precision value of 0.93 (Puyalnithi & Viswanatham, 2016). Decision trees have also been employed in prognostic endeavors related to cardiovascular diseases. In the context of right ventricular failure, which is a common concern following the implantation of a left ventricular assist device (LVAD), Wang et al. constructed a decision tree model trained on clinical records of 183 LVAD

recipients, encompassing parameters like right atrial pressure, heart rate, and white blood cell count. This model was employed to predict the need for right ventricular support and achieved an AUC of 0.87 (Wang et al., 2012).

2.8 Applications of Deep learning

Deep learning (DL), a prominent and burgeoning subfield within the domain of Artificial Intelligence (AI), has made significant strides and found application in addressing a multitude of intricate challenges. DL has achieved notable breakthroughs across various domains, encompassing tasks such as image recognition, classification, and segmentation (Krizhevsky et al., 2017), speech recognition (Sainath et al., 2015), genomics (Alipanahi et al., 2015), reconstruction of brain circuits (Helmstaedter et al., 2013), natural language comprehension (Collobert et al., 2011), and the recognition of heart sounds (Chen et al., 2016). This approach to machine learning endeavors to emulate the intricate processing of the human brain and is typically constructed upon multi-layered artificial neural networks. DL employs data to analyze complex hierarchical representations characterized by multiple levels of abstraction (LeCun et al., 2015). The extensive utilization of compute-intensive DL techniques has been made feasible through advancements in graphics processing units and cloud computing, ushering in a new era of DL capabilities (Raina et al., 2009).

Two prominent neural network algorithms within the DL framework are Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). For instance, Xiong et al. devised a DL-based model known as RhythmNet, amalgamating both CNN and RNN methodologies to undertake the classification and diagnosis of atrial fibrillation (AF) based on ECG data (Xiong et al., 2018). DL has found wide-ranging applications in the realm of cardiovascular diseases, encompassing conditions like arrhythmias, congestive heart failure, chronic heart failure, coronary artery disease, and

AF. In a recent study, Ali et al. introduced an ensemble DL-based smart healthcare monitoring system tailored for predictive heart disease detection (Ali et al., 2020). Additionally, Tison et al. harnessed deep neural networks to detect AF using smartwatch data (Tison et al., 2018).

Furthermore, DL's utility extends to the analysis of fundus photography within the field of cardiology studies (Son et al., 2020). For example, Poplin et al. leveraged retinal fundus images to predict cardiovascular risk factors, including age, gender, smoking status, as well as major adverse cardiac outcomes (Poplin et al., 2018). The DL model, trained with fundus photography data, achieved an AUC of 0.70, validated across two distinct cohorts.

2.8.1 Applications of Convolutional Neural Network

CNN comprises artificial multilayers that include a convolutional process responsible for extracting hierarchical features from raw input data. It is equipped with fully connected layers used as classifiers (Bizopoulos & Koutsouris, 2018). CNN has found extensive application in image processing and classification (Dilsizian & Siegel, 2018). Furthermore, CNN has been widely employed in the detection of arrhythmias using publicly available ECG databases, achieving an accuracy exceeding 75% (Dilsizian & Siegel, 2018). Hannun et al. utilized CNN to classify a broad spectrum of arrhythmia conditions, employing data from single-lead ECGs obtained from a substantial number of patients, and obtained an impressive AUC of 0.97 (Dilsizian & Siegel, 2018). This study exemplifies the promising potential of deep learning in clinical contexts, substantially enhancing the diagnosis and classification of arrhythmias.

DL techniques, utilizing CNN architectures, have also been explored for ECG data analysis aimed at detecting AF (Ping et al., 2020). ECG segmentation data, obtained through a signal conversion approach employing methods like the short-term Fourier

transform (STFT) or stationary wavelet transform (SWT), were used to train and test a deep CNN model for AF detection, achieving accuracies of 98.27% and 98.36% for STFT and SWT, respectively (Xia et al., 2018). Additionally, Zheng et al. proposed a method focusing on the Spectro-temporal data matrix of ECG signals, trained with a deep CNN model for AF detection (Zhao et al., 2018).

Photoplethysmogram data were also utilized for training CNN models to detect AF (Poh et al., 2018). Raw data from 180 hours of Photoplethysmography monitoring were employed, resulting in an impressive AUC of 0.99 (Gotlibovych et al., 2018).

In another study, an 11-layer CNN model was developed for diagnosing congestive heart failure using minimally pre-processed ECG signals, achieving an impressive accuracy of 98.97% (Acharya et al., 2019). DL techniques have also been applied to cardiac MRI images for tasks such as left-ventricle segmentation (Tan et al., 2017), left-ventricle/right-ventricle segmentation (Bai et al., 2018), and whole heart segmentation (Li et al., 2017). Furthermore, DL modeling with echocardiography images using CNN architectures has been employed to recognize 15 echocardiographic views (Madani et al., 2018). In another echocardiographic study, a CNN-based model classified echocardiographic images into 5 standard views with an accuracy of 98.1% (Madani et al., 2018).

2.8.2 Applications of Recurrent Neural Network

Recurrent neural network, or RNN constitutes a robust dynamic system that embodies a neural sequence model, adept at identifying temporal patterns within longitudinal data (Choi et al., 2017). RNN has demonstrated its capability to tackle intricate machine learning tasks, including the acquisition of linguistic phrases and the generation of natural language explanations for images and their respective regions (Cho et al., 2014). Choi et al. harnessed longitudinal electronic health record data to establish relationships among

time-stamped events, encompassing disease diagnoses and medication orders (Choi et al., 2017). This study serves as a testament to the promising utility of RNN in detecting incident heart failure, attaining AUC values of 0.777 and 0.883 when considering data collected over 12-month and 18-month observation windows, respectively.

2.8.3 Applications of Artificial Neural Network

Neural networks (NN), characterized by their adaptability, are versatile tools for analyzing diverse data types and solving a wide array of computational challenges, including identification, classification, and prediction (Bishop, 1995). These networks, comprised of numerous interconnected nodes distributed across multiple layers, bear a resemblance to the neurons in the human brain. It's noteworthy that NN was employed as a non-invasive method for diagnosing ischemic heart diseases and myocardial ischemia approximately two decades ago (Kukar et al., 1999). NN was utilized as an auxiliary approach to analyze clinical and follow-up data from heart failure (HF) patients, effectively assessing HF severity and HF type with accuracies of 77.8% and 84.73%, respectively (Guidi et al., 2014). Several investigations have drawn upon the UCI Machine Learning Repository, housing patients' clinical information such as age, sex, cholesterol levels, and resting blood pressure, to develop NN-based machine learning models for predictive HF diagnostics. These NN approaches have demonstrated commendable prediction accuracies, typically ranging between 80% and 90% (Wadhonkar et al., 2015).

Ruiz-Fernandez et al. (2016) embarked on an exploration of both supervised and unsupervised machine learning methodologies, analyzing data sourced from the Cardiovascular Foundation of Colombia. Their objective was to predict and classify risks associated with congenital heart surgery. Notably, the multilayer perceptron, a supervised artificial NN model, exhibited the highest prediction accuracy among the models

examined (Ruiz-Fernandez et al., 2016). Furthermore, Atkov et al. devised various NN-based models to assess diagnostic accuracies for coronary heart disease. These models explored different combinations of patient features, encompassing genetic factors, age, and coronary angiography data. The prediction accuracies achieved by these models spanned from 64% to 91% (Atkov et al., 2012).

2.9 Applications of Data balancing

Balancing the data can improve the prediction and help in reducing errors. For instance, the SMOTE is widely utilized in many works (Umer et al., 2022). In (Ishaq et al., 2021), the ETC was proposed where the RFC is used for the FS, and the SMOTE is utilized to make the data balances. SMOTE-based artificial neural network (ANN) was mentioned by (Waqar et al., 2021), whereas the Randomoversampler was employed for data balancing (Kibria & Matin, 2022), and the ML-based fusion approach consisting of adaptive boosting (AdaBoost) combined with a decision trees model was proposed for heart disease severity prediction. A hybrid RFC with a linear model (HRFLM) was proposed by (Mohan et al., 2019), and the proposed HRFLM managed to predict heart disease with an accuracy of 88.7%. Different intelligent ML models were used (Gupta et al., 2019; Haq et al., 2018). For instance, various ML models such as SVM, K-nearest neighbors and decision trees were used by (Haq et al., 2018) to predict heart disease, where the proposed hybrid intelligent system attained an accuracy of 88%. A machine intelligence framework for HD diagnosis was presented by (Gupta et al., 2019), where the factor analysis of mixed data is used to extract and derive features from the Cleveland dataset in order to train the ML prediction models. Further, a computational intelligence system for HD diagnosis was proposed by (A. Gupta et al., 2022), and the SMOTE was used to balance the unbalanced datasets. The proposed model has enhanced the accuracy compared to the literature techniques published in 2020 by 5.17%, with an accuracy of 97.37%. In contrast, an N2Genetic-nuSVM model proposed by (Abdar et al., 2019)

attained an accuracy of 93.08%. The teaching-learning-based optimization along with fuzzy C-means (TLBO-KM/FCM) model was suggested by (Dubey et al., 2021); the model outperformed the other models with an accuracy of 99.4%. Finally, the proposed work in (Ali et al., 2021) employed the RFC to predict the HD using the SOMTE for dataset balancing.

2.10 Applications of tuning the hyperparameter optimization

Tuning of the hyperparameter (HP) plays a significant role in better prediction accuracy. The grid search (GS) was used by (Ahmad et al., 2022) and (Gu et al., 2022) for the hyperparameter optimization (HPO), whereas the manual trails and the MGOHBO were employed for tuning the HP in (Tiwari et al., 2022) and (Shan et al., 2022), respectively. Further, the multi-objective particle swarm optimization (MOPSO) was used in (Asadi et al., 2021) to tune the parameters and feature selection in order to enhance the performance of the proposed RF for heart disease diagnosis. The results revealed the MOPSO-RF attained accuracy values of 85.21 and 88.26 for the Cleveland and Statlog datasets, respectively. The authors (Abdellatif, Abdellatef, Kanesan, Onn, et al., 2022) proposed an improved weighted RF to deal with the imbalance dataset problem on an algorithm level, and the algorithm was optimized using Bayesian optimization (BO). Finally, the verification test is considered a good indicator of the model's effectiveness across multiple datasets. The T-paired verification test was applied in (Fitriyani et al., 2020), and the proposed DBSCAN+ SMOTE-ENN+ eXtreme Gradient Boosting (XGB) was proposed to solve the cardiovascular prediction problem. The IG was employed to select the feature and the SMOTE edited nearest neighbour to balance the data. A two-step statistical significance test was presented in (Tama et al., 2020), a two-tier ensemble PSO-based FS model was used, and the hyperparameter was optimized using a GS. Finally, the hyperband for HPO is utilized in (Abdellatif, Abdellatef, Kanesan, Chow, et al., 2022) to optimize the ETC model for better heart disease detection.

2.11 Summary

The literature review identifies key advancements and research gaps in using ML models to predict CVD, focusing on addressing model stability, interpretability, and data imbalance issues. Existing models have made progress in CVD prediction but largely rely on simpler algorithms such as Decision Trees, Extra Trees Classifier, and Random Forests. In contrast, recent developments in ML, such as stack-ensemble learning and deep learning, offer more robust options. Many prior studies did not implement hyperparameter optimization (HPO), which is critical for improving model performance across various datasets. Additionally, handling imbalanced datasets remains a challenge, with methods like SMOTE being widely used but often introducing noise and bias, particularly in high-dimensional data.

Based on the limitations identified in previous studies, two methodologies are proposed to address the challenges in CVD prediction, one at the algorithm level and the other at the data level. The first methodology focuses on addressing class imbalance at the algorithm level. An Improved Weighted Random Forest is proposed, incorporating cost-sensitive learning to manage imbalanced datasets effectively. This approach also integrates supervised infinite feature selection for feature selection and Bayesian Optimization to optimize the model's performance. The aim is to enhance the accuracy and reliability of CVD detection and survival prediction.

The second methodology addresses data imbalance at the data level by proposing a stable and interpretable stack predictor for heart disease. This model combines a conditional variational autoencoder to balance the data distribution and solve the data imbalance on the data level with Bayesian Optimization for hyperparameter tuning. Integrating these techniques seeks to improve the predictive accuracy of CVD models while maintaining interpretability through model-interpretation tools. These methodologies, built upon the

gaps in the existing research, offer solutions to improve model performance, accuracy, and interpretability, providing a more robust framework for CVD prediction, which will be detailed further in Chapter 3.

In conclusion, the aforementioned research endeavors have been succinctly outlined and tabulated in Table 2.1. These investigations encompass various critical considerations, encompassing feature selection, data balancing techniques, hyperparameter optimization strategies, validation methodologies, statistical tests for model verification, interpretability of the models, and the specific datasets utilized in each study (Ahsan & Siddique, 2022).

Table 2.1: Summary of Literature Review

References	Method	FS	Balancing	HPO	Validation	Verification Test	Model Interpretation	Dataset
(Nilashi et al., 2020)	KNN+SOM+PCA+ Fuzzy SVM	SOM	×	×	×	×	×	Cleveland, Statlog
(Thanga Selvi & Muthulakshmi, 2021)	OANN (DBMRI-TLBO-ANN)	×	×	TLBO	10-fold cv	×	×	Cleveland
(Fitriyani et al., 2020)	DBSCAN+ SMOTE-ENN+ XGBoost	GI	SMOTE-ENN	×	10-fold cv	T-paired	×	Cleveland, Statlog
(Tama et al., 2020)	PSO-Two-tier ensemble	PSO	×	GS	10-fold cv	Two- statistical verification test	×	Z-Alizadeh Sani, Statlog, Cleveland, Hungarian
(Ishaq et al., 2021)	ETC	RF	SMOTE	×	×	×	×	HD clinical records

Table 2.1: Summary of Literature Review (continued)

(Kibria & Matin, 2022)	ADA+ DT	×	Randomoversampler	Not mentioned	×	×	×	Cleveland
(Haq et al., 2018)	Relief + LR	Relief	×	×	10-fold cv	×	×	Cleveland
(Waqar et al., 2021)	SMOTE-based ANN	×	SMOTE	×	×	×	×	Cleveland
(Mohan et al., 2019)	HRFLM	RF	×	×	×	×	×	Cleveland
(Ali et al., 2019)	Stacked SVM	SVM	×	HGSA	×	×	×	Cleveland
(Gupta et al., 2019)	MIFH	FAMD+RF	×	Not mentioned	holdout validation scheme	×	×	Cleveland
(Ahmad et al., 2022)	XGB	GBC evaluator	×	GS	5-fold cv	×	×	Kaggle dataset
(Dubey et al., 2021)	K-means+ fuzzy c-means	×	×	TLBO	10-fold cv	×	×	Cleveland, Statlog
(Almazroi, 2022)	DT	×	×	×	5-fold cv	×	×	HD clinical records

Table 2.1: Summary of Literature Review (continued)

(Umer et al., 2022)	CNN	×	SMOTE	×	10-fold cv	×	×	HD clinical records
(Abdar et al., 2019)	N2Genetic-nuSVM	PSO or GA	×	N2Genetic	10-fold cv	×	×	Z-Alizadeh Sani
(Tiwari et al., 2022)	Stacked ensemble method	×	×	×	10-fold cv	×	×	Heart disease (IEEE) dataset
(Shan et al., 2022)	MGOHBO-KELM	×	×	MGOHBO	10-fold cv	×	×	Cleveland, Statlog
(Vivekanandan & Narayanan, 2019)	DE-Cox regression	modified DE	×	×	×	×	×	Cleveland
(Ali et al., 2021)	RF	×	SMOTE	×	10-fold cv	×	×	Kaggle dataset
(A. Gupta et al., 2022)	C-CADZ	FAMD+BBA	SMOTE	Not mentioned	holdout validation scheme	×	×	Z-Alizadeh Sani
(Asadi et al., 2021)	RF	MOPSO	×	MOPSO	10-fold cv	Two- statistical verification test	×	Statlog, Cleveland, SPECT, PECTF, VA Long Beach, and Eric

Table 2.1: Summary of Literature Review (continued)

(Gu et al., 2022)	SSGNet	permutation importance	×	grid search	×	×	×	Heart Disease Cleveland
Proposed model 1	Inf-FS_s+BO+IWRF	Inf-FS_s	Algorithm-based	BO	10-fold cv	×	×	Statlog, Heart failure clinical records
Proposed model 2	SPFHD	SHAP	CVAE	BO	10-fold cv	Two- statistical verification test	SHAP Framework	Statlog, Cleveland, Heart failure clinical records Data Set, and Z-Alizadeh Sani

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter presents the methodology employed to achieve the research objectives. The CVD datasets are described in detail along with the data collection. The steps for data preprocessing are elaborated for use in classification methodology. Furthermore, the proposed methodologies (IWRF for data balancing on the algorithm level and CVAE for data balancing on the data level) for CVD detection are elaborated separately. In addition, the methodology of other detection methods, such as single and hybrid models [RFC, ETC, XGB and LGBM], data balancing techniques integrated with different classifiers such as SMOTE-RFC, is also discussed for comparison with the proposed methodologies technique. The performance metrics are discussed to assess the accuracy of detection techniques by evaluating the difference between predicted and actual values. Furthermore, the methodology of different optimization algorithms, namely: GA, PSO, RS, Hyperband, and BO, is presented to tune the hyperparameters of the developed methods to enhance CVD detection accuracy.

3.2 Data Collection and preprocessing

Data is the backbone of any analytical endeavor. Whether developing ML models, conducting statistical analyses, or generating business intelligence reports, the process invariably begins with collecting relevant data. However, raw data often comes with irregularities and imperfections. Thus, the subsequent cleaning and preparation steps become crucial to ensure that the data's potential is fully realized. After collection, data is seldom ready for immediate analysis. It might contain missing values, outliers, or erroneous entries. Data cleaning involves identifying and rectifying these issues. This might mean imputing missing values, correcting mislabeled data, or removing duplicates.

Once cleaned, the data needs to be transformed into a format suitable for analysis. This can involve various tasks such as encoding categorical variables, feature engineering, or normalization. One popular method for normalization is using the z-score. The formula for the Z-score normalization is:

$$z = \frac{x - \mu}{\sigma} \quad 3.1$$

Where x is the raw score, μ is the mean of the dataset, and σ is the standard deviation of the dataset. The Z-score essentially tells us how many standard deviations away a data point is from the mean. By converting data into z-scores, we ensure our dataset has a mean of 0 and a standard deviation of 1.

Z-score normalization is considered for several reasons, first is scale independence, where ML algorithms, particularly those that rely on distances like k-means clustering or k-nearest neighbors, can be sensitive to the scale of the features. Normalizing data using Z-scores will give all features the same scale, making these algorithms work more effectively. Second, Z-scores can be useful for identifying outliers. For instance, data points with Z-scores significantly greater than 3 or less than -3 could be considered outliers in many contexts. Third, improves convergence for optimization algorithms, especially those used in deep learning models; having features on the same scale can lead to faster convergence.

In cases like heart disease prediction, where datasets might contain features measured in different units (like age in years, cholesterol levels in mg/dL, and blood pressure in mmHg), using Z-score normalization ensures that no feature disproportionately influences the model's outcomes simply because of its scale. Multiple datasets are used to evaluate the proposed methodology discussed below.

3.2.1 Cleveland dataset

The Cleveland Heart Disease Dataset is one of the pioneering datasets in medical data science, especially concerning cardiovascular diseases (Janosi et al., 1988). Originating from the renowned UCI Machine Learning Repository, the Cleveland dataset has become emblematic, often a foundational stepping stone for many researchers and enthusiasts exploring heart disease prediction. The dataset's data was collected from real patients and aggregated several clinical parameters. It was conceived to discern patterns that influence the occurrence of heart disease. With its comprehensive features, the dataset provides a holistic view of the patient, capturing essential details from demographics to more intricate medical metrics. Over the years, numerous machine learning models have been trained on this dataset, making it a benchmark in heart disease research. The Cleveland dataset contains 303 samples with 14 features detailed in Table 3.1; the samples are divided into classes, 137 having CVD and 160 being non-CVD. However, one of its limitations is the potential imbalance in the class distribution. In some versions, there's a skewed representation of patients with and without heart disease, which can influence the performance of machine learning models. This imbalance might lead to more accurate models predicting the majority class while failing to identify the minority class effectively.

Table 3.1: Summary of Features in the Cleveland and Statlog Heart Disease Datasets: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature.

No. #	Attribute	Description	Type	Range
1	Age	Age in years	Numeric	29 to 77
2	Sex	Gender	Categorical	Female = 0, Male = 1
3	cp	Chest pain type	Nominal	Typical angina = 1, atypical angina = 2, non-anginal pain = 3, asymptomatic = 4
4	threstps	Resting blood pressure (mmHg)	Numeric	94 to 200 (mmHg)
5	chol	Serum cholesterol (mg/dl)	Numeric	126 to 564 (mg/dl)
6	fbs	Fasting blood sugar (value >120)	Categorical	False = 0, true = 1
7	restecg	Resting electrocardiographic	Categorical	normal = 0, ST-T wave abnormality = 1, Probable or definite left ventricular hypertrophy =2
8	thalach	Maximum heart rate	Numeric	71 to 202
9	exang	Exercise induced angine	Categorical	No = 0, Yes = 1
10	oldpeak	ST depression induced by exercise relative to rest	Numeric	0 to 6.2
11	slope	Slope of peak exercise ST segment	Categorical	Up-sloping = 1, Flat = 2, Down-sloping = 3
12	ca	Number of major vessels	Categorical	0 to 3
13	thal	Defect	Categorical	Normal = 3, Fixed = 6, Reversable = 7
14	Class	Predicted patient status	Categorical	Absence = 0, Presence = 1

3.2.2 Statlog

The Statlog (Heart) Dataset is another seminal dataset in cardiovascular research, distinguished by its amalgamation of data from diverse sources to offer a comprehensive set of records (*Statlog (Heart) Data Set*). Its fusion approach sets the Statlog dataset apart, making it a hybrid resource. While it carries similarities with the Cleveland dataset, the Statlog dataset presents its unique challenges and characteristics. It has been a cornerstone for many comparative studies that aim to discern the performance of models across

datasets of similar nature but different data distributions. Its utility has been demonstrated in various academic papers and projects, where it has consistently proven to be a valuable asset in understanding heart diseases' underlying patterns and risk factors. The Statlog dataset has the same features as Cleveland, with different ranges as shown in Table 3.1 and 270 samples; the samples are divided into classes, 120 having CVD and 150 being non-CVD. However, like the Cleveland dataset, Statlog also grapples with class imbalance. This disparity in representation can result in models that might be biased towards the more prevalent class. Researchers leveraging this dataset often employ techniques like resampling or synthetic data generation to address this imbalance and achieve more robust and generalizable models.

3.2.3 Z-Alizadeh Sani

The Z-Alizadeh Sani dataset is a contemporary addition to the collection of heart disease datasets, designed explicitly with the modern challenges of medical diagnostics in mind (Arabasadi et al., 2017). Unlike its predecessors, this dataset encompasses a broader range of clinical features, painting a more detailed portrait of patients and their cardiovascular health. Its design is focused on predicting the occurrence of coronary artery disease, a specific subset of heart diseases, making it highly specialized. With its extensive list of features, the Z-Alizadeh Sani dataset provides researchers and clinicians with deeper insights into the multifaceted nature of coronary artery diseases. It embodies the evolution of medical data, reflecting advancements in data collection and the growing understanding of heart diseases in the medical community. This dataset contains 54 features shown in Table 3.2 with a total of 303 samples; the samples are divided into classes as 216 have CVD and 87 have non-CVD. However, even with its advanced design, it isn't free from drawbacks. There's a known imbalance in the dataset, with more patients diagnosed with coronary artery disease than those without. This discrepancy can

influence machine learning endeavors, possibly leading to overfitting towards the majority class.

Table 3.2: Summary of Features in the Z-Alizadeh Heart Disease Dataset: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature.

No.	FT*	Attribute- Description	Type	Values
1	Demographic	Age	Numeric	30–86
2		Weight	Numeric	48–120
3		Length	Numeric	140-188
4		Sex	Categorical	M, F
5		BMI (Body Mass Index Kg/m2)	Numeric	18–41
6		DM (Diabetes Mellitus)	Categorical	Y, N
7		HTN (Hypertension)		
8		Current smoker		
9		Ex-smoker		
10		FH (Family History)		
11		Obesity	Categorical	Yes if MBI > 25, No otherwise
12		CRF (Chronic Renal Failure)	Categorical	Y, N
13		CVA (Cerebrovascular Accident)		
14		Airway disease		
15		Thyroid disease		
16		CHF (Congestive Heart Failure)		
17		DLP (Dyslipidemia)		

Table 3.2: Continue Summary of Features in the Z-Alizadeh Heart Disease Dataset: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature (continued)

18	Symptom and examination	BP (Blood Pressure mm Hg)	Numeric	30–86
19		PR (Pulse Rate ppm)	Numeric	50–110
20		Edema	Categorical	Y, N
21		Weak peripheral pulse		
22		Lung rales		
23		Systolic murmur		
24		Diastolic murmur		
25		Typical chest pain		
26		Dyspnea		
27		Function class		1, 2, 3, 4
28		Atypical		Y, N
29		Nonanginal chest pain		
30		Exertional chest pain		
31		Low Th Ang (low-Threshold angina)		
32	ECG	Q wave	Categorical	Y, N
33		ST elevation		
34		ST depression		
35		T inversion		
36		LVH (Left Ventricular Hypertrophy)		
37		Poor R-wave progression		
38		BBB		

Table 3.2: Continue Summary of Features in the Z-Alizadeh Heart Disease Dataset: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature (continued)

39	Laboratory and echo	FBS (Fasting Blood Sugar mg/dL)	Numeric	62–400
40		Cr (Creatine mg/dL)		0.5–2.2
41		TG (Triglyceride mg/dL)		37–1050
42		LDL (Low-Density Lipoprotein mg/dL)		18–232
43		HDL (High-Density Lipoprotein mg/dL)		15–111
44		BUN (Blood Urea Nitrogen mg/dL)		6–52
45		ESR (Erythrocyte Sedimentation Rate mm/h)		1–90
46		HB (Hemoglobin g/dL)		8.9–17.6
47		K (Potassium mEq/lit)		3.0–6.6
48		Na (Sodium mEq/lit)		128–156
49		WBC (White Blood Cell cells/mL)		3700–18,000
50		Lymph (Lymphocyte %)		7–60
51		Neut (Neutrophil %)		32–89
52		PLT (Platelet 1000/mL)		25–742
53		EF (Ejection Fraction %)		15–60
54		Region with RWMA	Categorical	0,1,2,3,4
55		VHD (Valvular Heart Disease)	Categorical	Normal, Mild, Moderate, Severe

3.2.4 Heart disease clinical records

The Heart Failure Clinical Records dataset, hosted on the reputable UCI ML Repository, embodies the symbiotic relationship between traditional medical research and contemporary data science techniques. This dataset is designed to predict mortality due to heart failure (Ahmad et al., 2017); it has gained traction among researchers and industry professionals for its comprehensive features and clinical relevance.

Heralding from actual clinical records, the dataset encapsulates many metrics pertinent to heart health. The dataset provides a panoramic view of factors influencing heart failure, from laboratory test results such as platelet counts to echocardiogram metrics like ejection fraction. Including lifestyle elements, such as smoking status, further enriches its breadth, making it a well-rounded resource for in-depth analyses. This dataset contains 299 samples divided into classes of 203 as non-CVD and 96 deceased because of CVD, and 12 features represented in Table 3.3.

Table 3.3: Summary of Features in the Heart Disease Clinical Records Dataset: A comprehensive overview detailing the names, descriptions, data types, and range of values for each feature.

No. #	Attribute	Description	Type	Range
1	Time	following up period	Numeric	4 to 285
2	Event (Target)	If the patient died in the following time	Categorical	0, 1
3	Sex	Male or Female	Categorical	Female = 1, Male = 0
4	Smoking	If the patient smokes	Categorical	0, 1
5	Diabetics	If the patient has diabetics	Categorical	0, 1
6	BP	If the patient has blood pressure problem	Categorical	0, 1
7	Anaemia	Decrease in red blood cell	Categorical	0, 1
8	Age	Age of the patient	Numeric	40 to 95
9	Ejection fraction	Percentage of blood leaving the heart at each concentration	Numeric	14 to 80
10	Sodium	Level of sodium in the blood	Numeric	114 to 148 (mEq/L)
11	Creatinine	Level of creatinine in the blood	Numeric	0.50-9.40 (mq/L)
12	Platelets	Platelets in the blood	Numeric	25.01-850 (Kiloplatelets/mL)
13	CPK	Level of CPK enzyme in the blood	Numeric	23-7861 (Mcg/L)
14	Time	following up period	Categorical	Absence = 0, Presence = 1

Table 3.4: The summary of the utilized datasets in the study.

Dataset Name	Num of features	Number of samples	Samples		The ratio between HD & Normal	Data distribution
			Normal	HD		
Cleveland	13	297	160	137	1: 1.17	Balanced
Statlog	13	270	150	120	1: 1.25	Balanced
Z- Alizadeh Sani	54	303	87	216	1: 0.4	Not Balanced
HD clinical records	12	299	203	96	1: 2.11	Not Balanced

3.3 Proposed Method for CVD detection and tackling the imbalanced issue on the algorithm level.

In this subsection, the methodology that describes CVD detection and overcoming the imbalanced problem on the algorithm level is elaborated. Figure 3.1 presents the flowcharts of the proposed method one (IWRF).

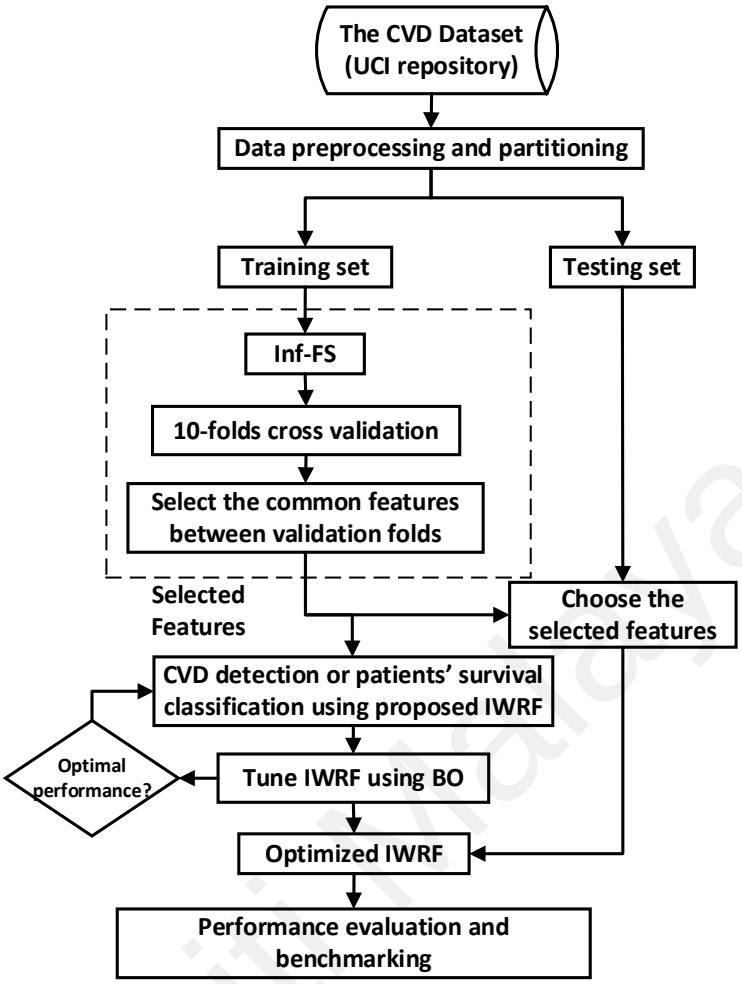


Figure 3.1: Flowchart of the proposed method one (IWRF)

3.3.1 Feature selection using infinite feature selection.

In the vast world of machine learning, feature selection is a cornerstone to constructing robust models. This process optimizes the input features and ensures that models aren't overburdened with irrelevant or redundant data, ultimately leading to better performance and more interpretable outcomes (Roffo et al., 2017). Enter Inf-FSs, a graph-based feature filtering approach has emerged as a leading method in feature selection. As delineated by (Roffo et al., 2020), Inf-FSs operates by considering all potential subsets of features and can work in both supervised and unsupervised forms. This method views the feature space as a fully connected, weighted, undirected graph $G = (V, E)$. In this depiction: nodes V represent all features, and edges E represent pairwise relationships between features.

If we envision G as an adjacency matrix A, then each component a_{ij} where $1 \leq I, j \leq n$, signifies the confidence degree that both nodes, namely v_i and v_j (features in the dataset context), are potential candidates for selection. The weight function, $\varphi(v_i,v_j)$, quantifies the value of each edge. In Inf-FSs, edge weight determination is an integration of class labels through the Fisher criteria and mutual information. Specifically, the weight function $\varphi(v_i,v_j)$ arises from three primary factors: first, Fisher criteria (h_i) quantifies how distinct two classes are for the i^{th} feature, expressed in Equation 3.2:

$$h_i = \frac{|\mu_{i,1} - \mu_{i,2}|^2}{\sigma_{i,1}^2 + \sigma_{i,2}^2} \tag{3.2}$$

Here, μ and σ represent the mean and standard deviation of the i^{th} attribute for a given class, respectively. Second, Normalized Mutual Information (m_i): A measure indicating the reduction in uncertainty about each class based on the knowledge of a feature vector, expressed in Equation 3.3:

$$m_i = \sum_{y \in Y} \sum_{z \in f_i} p(z,y) \log\left(\frac{p(z,y)}{p(z)p(y)}\right) \tag{3.3}$$

Where Y represents class labels, and $p(z,y)$ denotes the joint probability distribution. Third, calculate the Normalized Standard Deviation (σ_i) for a feature normalized to [0,1] using the maximum standard deviation across all features. Subsequently, a linear combination of these factors provides a score s_i for each feature.

$$s_i = h_i\alpha_1 + m_i\alpha_2 + \sigma_i\alpha_3 \tag{3.4}$$

The coefficients α_k, limited to [0,1] and summing up to 1, can be tuned experimentally. Finally, the adjacency matrix A is constructed using:

$$A(i, j) = \varphi(\overrightarrow{v_i}, \overrightarrow{v_j}) = s_i s_j \tag{3.5}$$

To illustrate the working of the Inf-FSs on your heart disease dataset. For instance, the heart disease dataset contains 13 features such as age, gender, chest pain type, resting

blood pressure, cholesterol levels, etc. These features are used to predict the target variable presence or absence of heart disease. As always, begin with data normalization and cleaning to ensure the feature values are on a comparable scale. After scaling, the three metrics is calculated, the Fisher criteria h_i for each feature, which quantifies the separation between the classes (presence/absence of heart disease) based on each feature. The mutual information m_i for each feature indicates how much information about the class labels each feature provides. The normalized standard deviation σ_i is calculated for each feature. The computed metrics for each feature determine the score si for each feature using Equation 3.4. Once each feature's score is determined, the adjacency matrix *A* is constructed using Equation 3.5. After constructing the adjacency matrix, rank the features based on their scores in the matrix. Higher scores suggest higher relevance and less redundancy concerning other features. Use cross-validation to determine the optimal mixing coefficients α_k. Depending on the CV results, tweak these coefficients for the best performance. For example, Inf-FSs will rank features like 'chest pain type' or 'resting blood pressure' higher due to their direct correlation with heart diseases.

3.3.2 Improved Weighted Random Forest

Random Forest Classification (RFC) is a tree-based ensemble learning technique that is an extension of Breiman's ensemble of decision trees (Breiman, 2001). RFC has gained significant popularity in recent years due to its superior performance, simplicity of implementation, and low computational cost. A random forest is basically an ensemble of tree predictors $\mathrm{fn}(\mathrm{X};\theta n)$. Each tree operates based on a series of conditional statements, often visualized in a decision tree format as per graph theory. The fundamental principle guiding the RFC is the bootstrapping methodology coupled with aggregation, colloquially known as bagging. Given a dataset of size *N*, RFC generates multiple bootstrap samples, each constructed by randomly choosing *N* instances from the

original dataset, allowing for replacements. This ensures diversity as some data points may be sampled multiple times, while others might be missed entirely.

For each bootstrap sample, RFC builds a decision tree. However, instead of considering all predictors at every node as in a conventional decision tree, RFC considers only a random subset of predictors. This subset typically comprises $\sqrt{M}$ predictors for classification tasks and $M/3$ for regression, where M is the total predictor count. Such randomness not only promotes tree diversity but also bolsters the forest's robustness. RFC trees split nodes based on impurity measures (Bashar et al., 2020). For classification, Gini impurity or entropy is prevalent. The Gini impurity for a node t is calculated as in Equation 3.6:

$$Gini(t) = 1 - \sum_{i=1}^{c} p(i|t)^2 \quad 3.6$$

Where $p(i|t)$ is the proportion of samples that belong to class i for node t. The model's training aims to optimize these conditional statements to best fit the data. This is essentially an optimization task targeting either the minimization of the misclassification rate or the maximization of a given impurity criterion, such as Gini impurity. The objective function can be denoted as in Equation 3.7.

$$\text{Objective function} = \min_{\theta} \sum_i I(\text{fn}(x_i; \theta n) \neq y_i) \quad 3.7$$

where I is an indicator function outputting 1 if prediction $fn(xi;\theta n)$ mismatches the actual label yi, and 0 otherwise, uniquely, in RFC, each tree split contemplates only a random subset of features, introducing the characteristic "randomness." This tactic effectively curbs overfitting. Often, trees are also confined by certain parameters like maximum depth or maximum leaf nodes. Another advantageous RFC feature is the Out-

of-Bag (OOB) samples. These are data points excluded from a particular tree's training and serve to assess the tree's performance.

Predictions in RFC stem from a democratic process: each tree in the forest "votes" for a class. The class receiving the majority becomes the final prediction. For a more formal representation described in Equation 3.8.

$$Ypred = mode(\{\, \mathrm{f1(X;}\theta 1), \mathrm{f2(X;}\theta 2), \ldots., \mathrm{fn(X;}\theta n) \qquad 3.8$$

Where $\mathrm{fn(X;}\theta n)$ is the prediction of n^{th} tree, N symbolizes the total tree count. The mode function yields the class with the highest vote frequency.

In imbalanced data classification, RF classifiers tend to be biased in the direction of the major class since standard RF treats both classes equally. In addition, there is a substantial chance that a bootstrap sample has few or no instances of the minority class, leading to a tree with low performance in predicting the minority class. However, several studies have shown that a weighted RF can deliver better prediction results. For this reason, this study presents an Improved Weighted Random Forest (IWRF).

In the traditional RFC, each bootstrap sample from the dataset is chosen randomly, which can perpetuate class imbalance in each sample. In the custom version, we introduce controlled sampling. For each bootstrap sample of size *N*, we ensure that a specified proportion of *p* is selected from the minority class, with the remainder coming from the majority class. The parameter *p* can be defined for each bootstrap, either randomly within a specified range or through optimization methods.

For example, selecting *p* within the range [0.3, 0.5] ensures that each bootstrap sample has at least 30% and at most 50% of instances from the minority class. This helps preserve the majority class's diversity while also giving adequate representation to the minority class.

Imagine a dataset of 100 instances with an imbalanced ratio of 0.4:1 (minority: majority). Traditionally, in RFC, if you were to draw a bootstrap sample of 100 instances, it could contain approximately 27 instances from the minority class and 73 from the majority class, due to random selection. While for the IWRF: If *p* is set to 0.4, then 40% of the instances in each bootstrap sample should be from the minority class. Hence, for a bootstrap sample of size 100, you would deliberately select 40 instances from the minority class. The remaining 60 instances would be drawn randomly from the majority class. This ensures a consistent representation of the minority class in each bootstrap sample, giving it a stronger voice in the model's decision-making process.

Moreover, the IWRF assigns a weight for each class a higher weight for the minor class. The class weight in the random forest can be computed using the inversely proportional class frequencies in the training dataset. The class weights are presented as the following in Equation 3.9.

$$CW_1 = \frac{M}{2M_1} \;\&\; CW_2 = \frac{M}{2M_2} \tag{3.9}$$

Where *M* presents the total number of samples in the dataset, M_1 and M_2 show the number in major and minor classes. We assign a new coefficient, the weighting factor (α), to compute class weights. Thus, the class weights will be calculated as in Equation 3.10:

$$CW_1 = \alpha_1 \frac{M}{2M_1} \;\&\; CW_2 = \alpha_2 \frac{M}{2M_2} \tag{3.10}$$

Where α_1 and α_2 are the weighting factor for major and minor classes, respectively, α_1 and α_2 vary in a range from [0, 1] with default values M_1/M_2 and one for α_1 and α_2. To ensure that CW_2 is always greater than CW_1 to have a heavier penalty on misclassifying

the minor class, the weighting factor is subjected to the constrain as shown in Equation 3.11:

$$\frac{\alpha_1}{M_1} < \frac{\alpha_2}{M_2} \tag{3.11}$$

The RF algorithm incorporates class weights in two places. Class weights are used in the tree induction technique to weight the Gini criteria for detecting splits. Class weights are again considered at each tree's terminal nodes. A weighted majority vote establishes each terminal node's class prediction. This adds weight to influence the decision-making process in tree construction. In Gini impurity calculations as presented in the Equation 3.12.

$$Gini(t) = 1 - \sum_{i=1}^{c} w_i p(i|t)^2 \tag{3.12}$$

where w_i is the weight of class i, the misclassification of a minority class instance results in a heightened penalty due to its higher associated weight. This can drive the decision tree to prioritize the correct classification of minority class samples.

For instance, suppose without weights, where a node has 5 instances belonging to not having CVD (majority) and 3 to having CVD (minority). If a split results in 4 from not having CVD and 3 from having CVD in one child node, it may seem acceptable from an impurity standpoint without weights. However, with class weights, the decision to make such a split becomes more expensive due to the higher penalty for misclassifying the minority class. The tree may optimize for a different split to reduce this penalty. Also, when determining the class of a terminal node, weights again play a role. A node populated with many majority class (not having CVD) instances might still predict the minority class (having CVD) if the accumulated weight of the minority instances surpasses that of the majority.

In the context of IWRF, given the introduced class weights, each tree's vote isn't just a simple vote anymore. It’s a weighted vote, where the weight is derived from the importance of the class, especially designed to tackle imbalances in the dataset.

For a tree that predicts a sample to belong to a particular class, the vote it casts for that class is multiplied by the weight for that class, as in Equation 3.13:

$$Weighted\ Vote_i = \text{fn}(X; \theta n) * CW_i \quad 3.13$$

Where CW_i is the weight for the class i. The final prediction is then based on the sum of weighted votes for each class across all trees, as expressed in Equation 3.14.

$$\begin{cases} 1\ if\ \sum_{n=1}^{N} (\text{fn}(X; \theta n) * CW_1) > \sum_{n=1}^{N} (\text{fn}(X; \theta n) * CW_2) \\ 0\ otherwise \end{cases} \quad 3.13$$

3.3.3 Bayesian Optimization

Bayesian Optimization (BO) is an iterative algorithm widely recognized for its proficiency in Hyperparameter Optimization (HPO) challenges (Snoek et al., 2012). The foundation of BO is grounded on two pivotal components: an acquisition function and a surrogate model. This surrogate model, which is often probabilistic, encapsulates and emulates the behavior of the objective function based on observed evaluations.

Upon establishing the surrogate model's predictive distribution, the acquisition function is critical when choosing prospective hyperparameter combinations. It masterfully balances exploration (probing untested regions) and exploitation (focusing on promising regions with expected optimal outcomes). By integrating these dual strategies, Bayesian tuning efficiently identifies the most likely optimal regions and ensures that potentially superior configurations in lesser-known regions are not overlooked (Hazan et al., 2017).

In the context of BO, a standout surrogate model is the Tree-structured Parzen Estimator (TPE) introduced by (Bergstra et al., 2011). At the core of TPE is creating two probability density models: $l(x)$ representing better outcomes and $g(x)$ denoting poorer outcomes. Importantly, these models are not strict likelihoods but instead capture the underlying densities of hyperparameters based on their observed performances. A predefined threshold, typically a percentile such as y^* of the observed objective values, helps delineate 'good' from 'bad' outcomes. Central to TPE is the non-parametric approach of Parzen Windows, used to estimate the probability density function of a random variable. In essence, this method places a 'window' or kernel around each data point in its sample. The shape and size of this window dictate the estimated density. Aggregating these windows over all data points furnishes a comprehensive approximation of the density function, allowing it to flexibly adapt to observed data without any predefined form for the underlying distribution.

Guided by the ratio between $g(x)$ and $l(x)$, the acquisition function then selects new configurations for evaluation, as emphasized by (Elshawi et al., 2019). Modern implementations of BO have enhanced parallelization capabilities, making it feasible to evaluate multiple configurations simultaneously. Thus, the earlier belief that BO is inherently sequential and hard to parallelize has been countered. Within a relatively limited number of iterations, BO can zone in on optimal or near-optimal hyperparameter values, minimizing computational overhead (DeCastro-García et al., 2019).

For instance, when tuning the IWRF classifier, hyperparameters such as α_1 and α_2 (the coefficients for class weights), n_estimators, max_depth, and max_features, and other hyperparameters are selected and set to the search space. BO-TPE could suggest starting with a medium number of trees, deep trees, and a high number of features for the first iteration. Based on the performance of this configuration (using f1 score as the objective

metric), the surrogate model is updated. In subsequent iterations, TPE might recommend exploring shallower trees or fewer features, iterating and refining the recommendations until an optimal configuration emerges or until a predefined stopping criterion is met.

Given that $l(x)$ is the likelihood of x based on good results and $g(x)$ is the likelihood of x based on poorer results, the Expected Improvement (or acquisition function) can be defined as the ratio as defined in Equation 3.14:

$$I(x) = \frac{g(x)}{l(x)} \tag{3.14}$$

We're looking to sample the next hyperparameter x at a point where the expected improvement is maximized. With a given threshold y^* to segregate good and bad results, we use all observed hyperparameters x with a function value better than y^* to model $l(x)$ and all others to model $g(x)$. The generative model can be expressed with Parzen windows as shown in Equation 3.15.

$$p(x|y, D) = \begin{cases} l(x), & y < y^* \\ g(x), & y \geq y^* \end{cases} \tag{3.15}$$

Where D represents the search space of the hyperparameters, the efficacy of BO-TPE is often contingent on the characteristics of the problem domain. For high-dimensional search spaces, where hundreds of hyperparameters need evaluation, the efficiency advantage of BO may diminish. There might also be interdependencies between hyperparameters, which TPE captures naturally through its tree structure, but this could add complexity. Another constraint is the computational cost of updating and querying the surrogate model, especially when the dataset is large.

The objective function in hyperparameter tuning is often the model's performance on a validation set using a given hyperparameter configuration. As new configurations are evaluated, the objective function's knowledge grows. For an IWRF, the F1-score is taken

as a metric. The fitness function, which can be synonymous with the objective function in this context, is updated with each new evaluation, providing a richer understanding of the hyperparameter space and informing subsequent iterations of the BO process.

In the context of IWRF hyperparameter tuning, our objective function is the validation error ε concerning the hyperparameters x. It can be represented as in Equation 3.16:

$$\varepsilon = f(x) \tag{3.16}$$

Where f is the IWRF classifier's performance dependent on the hyperparameter configuration x, the steps of BO-TPE can be summarized as follows:

1. **Dataset Partitioning**: The dataset is divided into training, validation, and testing sets. The test set is kept untouched until the final evaluation.
2. **Defining Search Space**: Identify the hyperparameters to tune. The hyperparameters for the proposed IWRF and other selected models for comparison are set in Table 3.4.
3. **Initial Sampling**: Start with a few random hyperparameter combinations to obtain initial data points, which can guide the TPE model in subsequent iterations.
4. **Constructing the TPE Model**:
    - Based on a predefined threshold (y^*), segregate the observations into "good" and "bad" outcomes.
    - Construct two probability models: *l(x)* for good outcomes and *g(x)* for bad outcomes. These models are built using Parzen Windows, a non-parametric approach.
5. **Acquisition Function Calculation**: For TPE, the acquisition function is defined as the ratio between the likelihoods given by the two models *g(x)* and *l(x)*, as in Equation 3.14. This acquisition function will suggest areas with higher chances of improvement.

6. **Selection of Next Point**: Choose the hyperparameter combination that maximizes the acquisition function, guiding where to evaluate next.
7. **Update the Model**: Evaluate the selected hyperparameter combination using cross-validation on the training set. Based on this new data point (hyperparameter configuration and its corresponding performance), update the $l(x)$ and $g(x)$ models.
8. **Iteration**: Steps 4 to 7 are repeated for a predefined number of steps or until convergence.
9. **Parallelization**: Modern TPE implementations can evaluate multiple hyperparameter configurations in parallel. This speeds up the process by allowing simultaneous evaluations of the objective function, which is especially useful for computationally intensive tasks with many hyperparameter configurations.
10. **Final Model Selection and Evaluation**: Once TPE iterations are complete, select the hyperparameter combination with the best performance on the validation set. Re-train the IWRF and other ML models with these optimal hyperparameters on the combined train-validation set. Evaluate its performance on the independent test set to get an unbiased estimate of its capability. The flowchart of the BO-TPE process is illustrated in Figure 3.2.

Table 3.5: Model Hyperparameter Exploration: A Comprehensive Table of Hyperparameters Selections, Types, and Ranges

Model	Selected Hyperparameter	Type	Search Space
SVM	Kernel	Categorical	['linear',' sigmoid',' poly',' rbf']
	C (penalty par.)	Continuous	[0.1,20]
	gamma (kernel paramter)		[0.05,0.2]
kNN	n_neighbors	Discrete	[1,20]
	weight	Categorical	[‘uniform’, ‘distance’]
LR	penalty	Categorical	[‘L1’, ‘L2’]
	C	Continuous	[0.1,20]
	solver	Categorical	['liblinear', 'lbfgs', 'sag']
SGD	Alpha	Continuous	[0.0001,0.1]
	penalty	Categorical	[‘L1’, ‘L2’]
ETC & RFC	n_estimators	Discrete	[10, 100]
	min_samples_splits		[2, 7]
	min_samples_leaf		[1, 7]
	max-depth		[5, 35]
	max-features		[1, 12]
	criterion	Categorical	['gini', 'entropy']
LGBM & XGBoost	n_estimators	Discrete	[10, 100]
	Learning rate	Continuous	[0.01, 0.5]
	subsample		[0.5, 1]
	colsample_bytree		[0.5, 1]
	max_depth	Discrete	[5, 30]
GBC	learning_rate	Continuous	[0.1, 0.5]
	subsample		[0.5, 1]
	n_estimators	Discrete	[10, 100]
	max_depth		[3, 15]

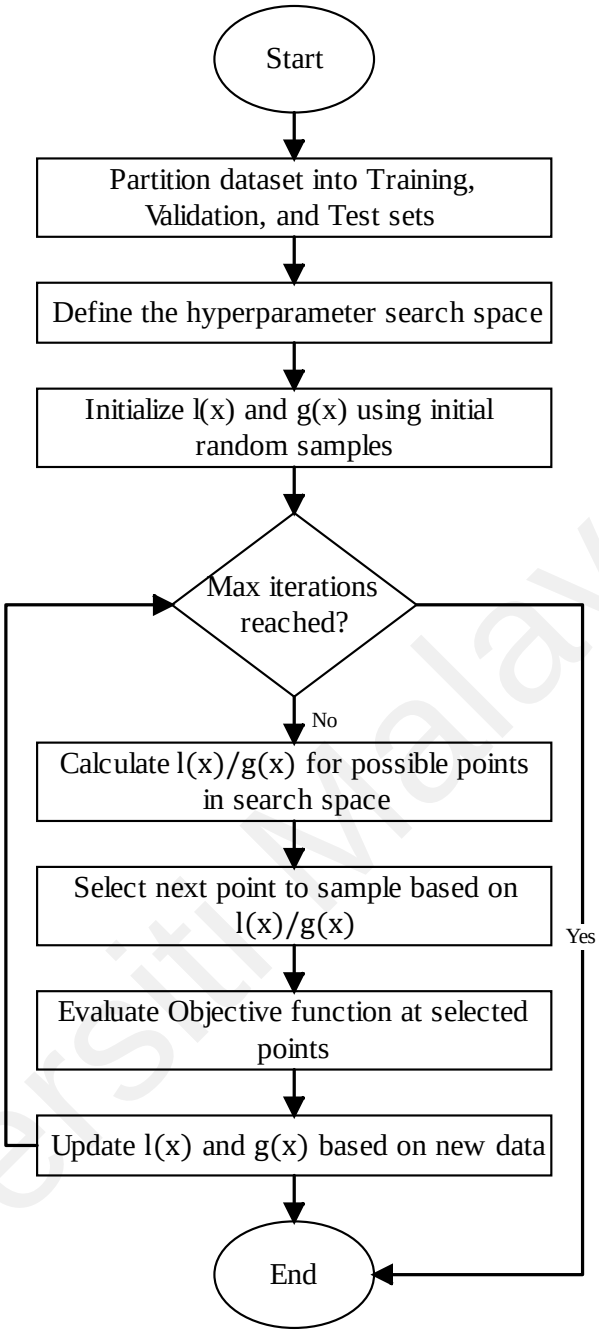


Figure 3.2: BO-TPE A Graphical Representation of Bayesian Hyperparameter Tuning with Tree Parzen Estimator of ML Models

3.3.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a nature-inspired optimization technique grounded on the social behavior of birds and fish. It has gained traction for its efficacy in HPO challenges (Kennedy & Eberhart, 1995). PSO's key components include a swarm of particles and a fitness function, which acts as the objective function that the algorithm tries to optimize.

In PSO, each particle represents a potential solution in the hyperparameter space. The particles iterate through the space, adjusting their positions based on their own experience and the experience of their neighbors. This can be viewed as an amalgamation of exploration and exploitation, akin to the dual strategies employed in Bayesian Optimization. PSO efficiently identifies promising regions in the hyperparameter space while also probing less-explored areas, ensuring that optimal configurations are not missed (Bonyadi & Michalewicz, 2017).

Central to PSO is the concept of "velocity," which determines how particles adjust their positions in the hyperparameter space. The velocity is updated based on cognitive and social components, usually guided by the best-known positions of the particle and its neighbors. The mathematical formula for velocity adjustment typically includes inertia, cognitive, and social components, each weighted by a factor are expressed in Equations 3.17, 3.18. In PSO, each particle i has a position X_i in the hyperparameter space and a velocity V_i. These are updated according to:

$$V_i(t+1) = w.V_i(t) + c_1.r_1.\left(P_{best,i} - X_i(t)\right) + c_2.r_2.(G_{best} - X_i(t)) \quad 3.17$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad 3.18$$

Where

- w is the inertia weight
- $c1$, $c2$ are cognitive and social scaling factors
- $r1$, $r2$ are random numbers in [0, 1]
- $P_{best,i}$ is the best-known position for particle i
- G_{best} is the best-known global position
- t denotes the iteration number

The inertia weight w controls the impact of the previous velocity, while c_1 and c_2 determine how much the particle is influenced by its best and the swarm's global best positions, respectively. In this study, w, c_1 and c_2 are set to be 0.9, 1.5, and 2 respectively.

When tuning hyperparameters for an IWRF classifier, PSO could initialize particles randomly across the hyperparameter space, covering α_1 and α_2 (the coefficients for class weights), n_estimators, max_depth, max_features, and other relevant parameters. The fitness function can be the model's performance on a validation set using F1-score metrics. As particles iterate, they hone in on the best-performing hyperparameter configurations based on the fitness function.

In the context of hyperparameter tuning, let $f(x)$ be the performance metric dependent on hyperparameter configuration x. The objective function, or fitness function, to be optimized can be denoted as in Equation 3.19:

$$\varepsilon = f(x) \tag{3.19}$$

Finally, the steps of PSO can be summarized as follows:

1. **Dataset Partitioning**: Same as in BO-TPE.
2. **Defining Search Space**: Use the same hyperparameters and their search space as BO-TPE, detailed in Table 3.4.
3. **Initialization Sawrm**: Begin with a swarm $S=[p^{(1)},p^{(2)},\ldots,p^{(m)}]$, where m is the number of particles. Each particle $p(i)$ represents a unique set of hyperparameters like $[\alpha_1,\alpha_2,n\_\text{estimators},\ldots]$.
4. **Fitness Function $F(p)$**: Measure the performance of a model with a given set of hyperparameters p. This is the F1-score of an IWRF classifier with hyperparameters p.

5. **Calculate Velocities**: Each particle *i* has a velocity V_i. Update the velocities based on the best-known positions of the particles and the global best position G_{best}. The updated equation is:

$$V_i(t+1) = w.V_i(t) + c_1.r_1.\left(P_{best,i} - p_i(t)\right) + c_2.r_2.(G_{best} - p_i(t))$$

6. **Update Positions**: Update the position of each particle using its new velocity:

$$p_i(t+1) = p_i(t) + V_i(t+1)$$

7. **Evaluate and Update Best Positions**: Calculate $F(p)$ for each particle and update the best-known positions $P_{best,i}$ and G_{best}.
8. **Iteration**: Repeat steps 3-6 for a set number of iterations or until convergence to find a particle p^* that maximizes $F(p)$.
9. **Parallelization**: Modern PSO implementations support parallel evaluations, which is beneficial for computationally expensive tasks.
10. **Final Model Selection and Evaluation**: Select the hyperparameter configuration corresponding to the global best position. Re-train the model using this configuration on the combined training and validation sets and evaluate its performance on the test set. Figure 3.3 illustrates the flowchart of PSO.

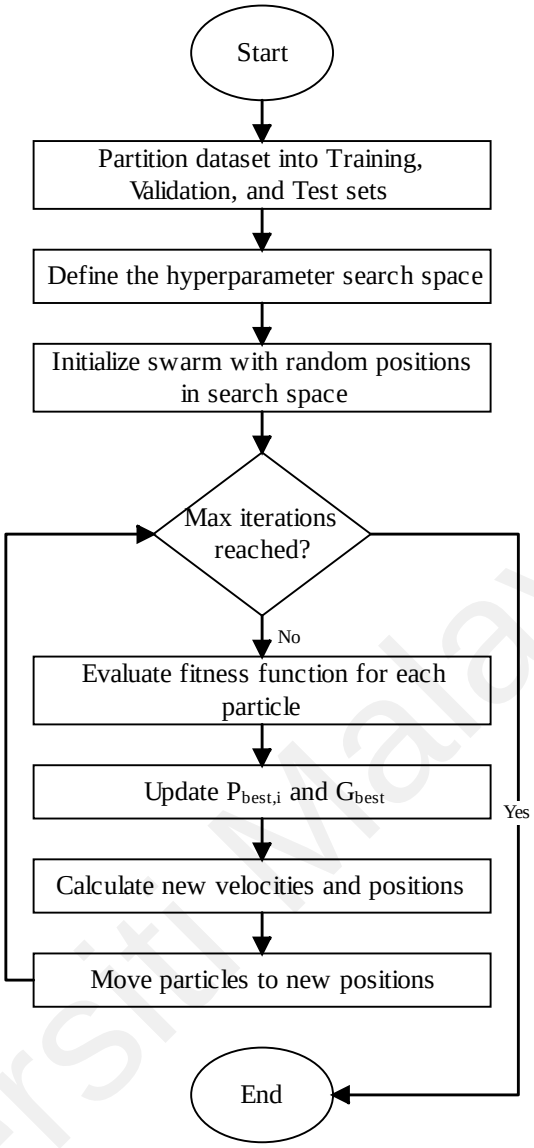


Figure 3.3: PSO Workflow for Hyperparameter Tuning of ML Models

3.3.5 Genetic Algorithm

Genetic Algorithm is a nature-inspired optimization technique that simulates the process of natural selection to find optimal solutions. It has been successfully applied to several HPO problems (Arabasadi et al., 2017; Deekshatulu & Chandra, 2013; Goldberg et al., 1989; Sun et al., 2020). At its core, GA uses a population of individuals (solutions), where each individual represents a possible hyperparameter configuration for the problem at hand. GA performs selection, crossover (recombination), and mutation operations to evolve the population towards better solutions over time. This can be seen as a balance between exploration and exploitation, much like in Bayesian Optimization and PSO.

In GA, each individual i has a chromosome X_i that encodes a possible solution in the hyperparameter space. These individuals are evolved using genetic operations, typically based on the fitness of each solution (Booker et al., 1989). The GA operations can be mathematically formalized as:

Selection: A fitness-proportional selection mechanism could be employed, where the probability $P_{\text{select}}(i)$ of selecting an individual i is proportional to its fitness $F(X_i)$ as expressed in Equation 3.20.

$$P_{select}(i) = \frac{F(X_i)}{\sum_j F(X_j)} \quad 3.20$$

Where the j is a variable used in the summation $\sum$, it iterates over all the individuals in the population, to sum up their fitness values. Essentially, the denominator $\sum_j F(X_j)$ calculates the total fitness of all individuals in the population to normalize the probability of selection.

Crossover: For two parent chromosomes X_1 and X_2, the crossover operation generates two offspring Y_1 and Y_2 as shown in Equation 3.21.

$$Y_1 = \alpha X_1 + (1 - \alpha)X_2,\ Y_2 = \alpha X_2 + (1 - \alpha)X_1 \quad 3.20$$

Where α represents the crossover rate. Mutation: This operation introduces small random changes in chromosome X, denoted as Mutate(X).

In the context of GA, let's consider a simplistic chromosome as a string $x = [x_1, x_2, \ldots, x_n]$ where each x_i is a hyperparameter (like α_1, α_2, n_estimators, etc.). The fitness function $F(x)$ would be the model's performance with these hyperparameters. We would initiate a population $P=[x^{(1)}, x^{(2)}, \ldots, x^{(m)}]$ where m is the total number of these individual solutions you are considering simultaneously at any given iteration and evolve this population over multiple generations until we find a chromosome x^* that maximizes $F(x)$. The objective

function to optimize could be the IWRF model's performance on a validation set using the F1-score metric. Let $f(x)$ be the performance metric, the fitness function can be expressed as in Equation 3.21:

$$\varepsilon = f(x) \tag{3.21}$$

The steps of the GA can be summarized in the following steps:

1. **Dataset Partitioning**: Similar to BO-TPE and PSO, the data is divided into training, validation, and testing sets.
2. **Defining Search Space**: The hyperparameters and their search space are similar to those in BO-TPE and PSO; refer to Table 3.4.
3. **Initialization**: Start with a population $P=[x^{(1)}, x^{(2)},\ldots,x^{(m)}]$ where m is the number of individual solutions (chromosomes) you start with. Each $x(i)$ is a unique set of hyperparameters, e.g., $x^{(1)}=[\alpha_1,\alpha_2,n\_\text{estimators},\ldots]$.
4. **Fitness Function $F(x)$**: The fitness function evaluates the performance of a model with a given hyperparameter set x. For instance, this could be the F1-score of an IWRF classifier with hyperparameters x.
5. **Selection**: Select pairs of chromosomes based on their fitness for the crossover operation.
6. **Crossover**: In crossover, pairs of parent chromosomes produce child chromosomes. For example, consider two parent chromosomes $x^{(i)} = [x_1^i, x_2^i, \ldots., x_n^i]$ and $x^{(j)} = [x_1^j, x_2^j, \ldots., x_n^j]$. A simple one-point crossover might produce one child as follows:
    1. Child 1: $[x_1^i, x_2^i, \ldots., x_k^i, x_{k+1}^j, \ldots., x_n^j]$

Here, k is a randomly selected crossover point.

7. **Mutation**: After crossover, the mutation is applied to the offspring with a certain probability. Mutation slightly alters one or more hyperparameters in a chromosome. For example, an offspring $[x_1,x_2,\ldots,x_n]$ might be mutated to $[x_1,x_2+\Delta,\ldots,x_n]$, where Δ is a small random change.
8. **Next Generation**: The children produced by crossover and mutation replace the least fit individuals in the population, creating the next generation of solutions.
9. **Iteration**: Steps 3-6 are repeated for a predefined number of generations or until a stopping criterion is met, such as finding a chromosome x^* that maximizes $F(x)$.
10. **Parallelization**: Modern GA implementations also allow for parallel evaluations of the fitness function, particularly useful for computationally demanding tasks.
11. **Final Model Selection and Evaluation**: At the end of the GA run, the individual with the highest fitness is selected. The model is retrained using this hyperparameter configuration on the combined training and validation sets, and its performance is evaluated on the test set. Figure 3.4 shows the flowchart of GA.

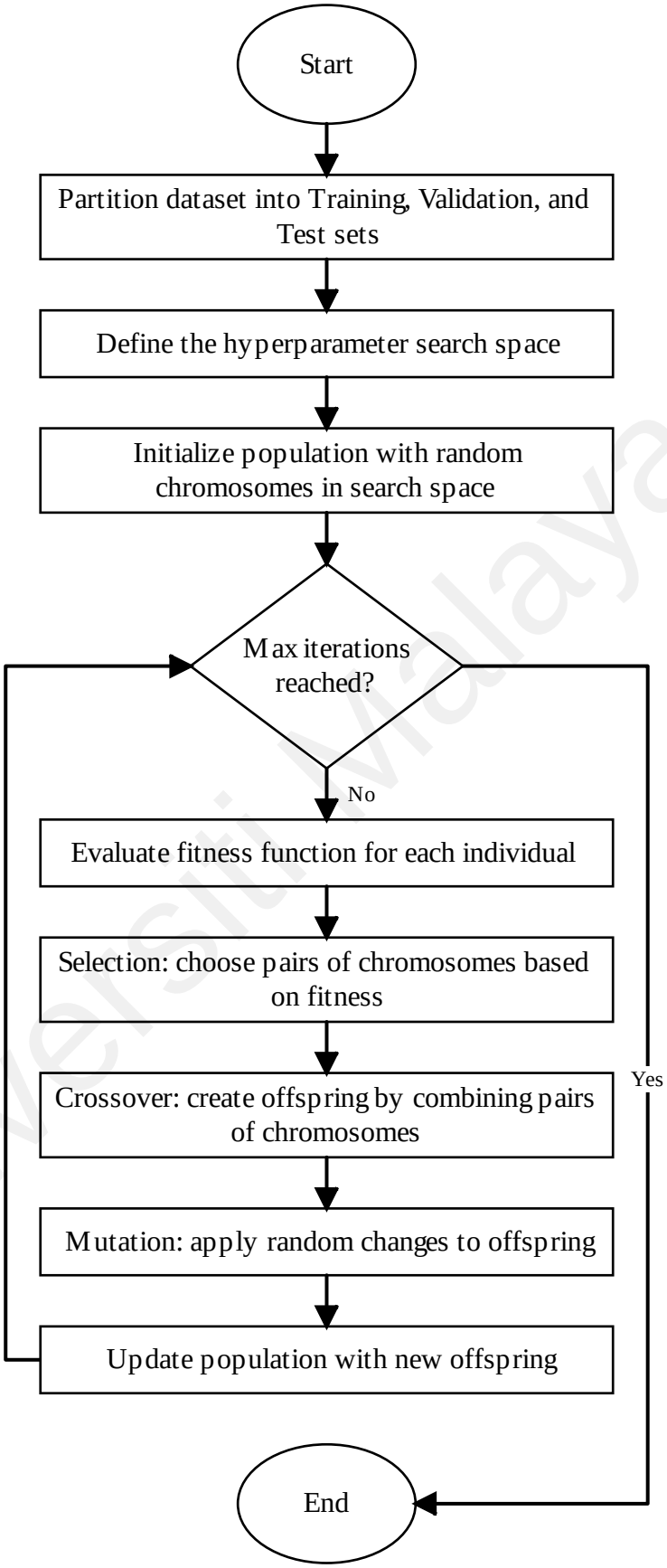


Figure 3.4: GA Workflow for Hyperparameter Tuning of ML Models

3.4 Proposed Method for CVD detection and tackling the imbalanced issue on the data level.

In this subsection, the methodology that describes CVD detection and overcoming the imbalanced problem on the data level is elaborated. Figure 3.5 presents the flowcharts of the proposed method two (SPFHD).

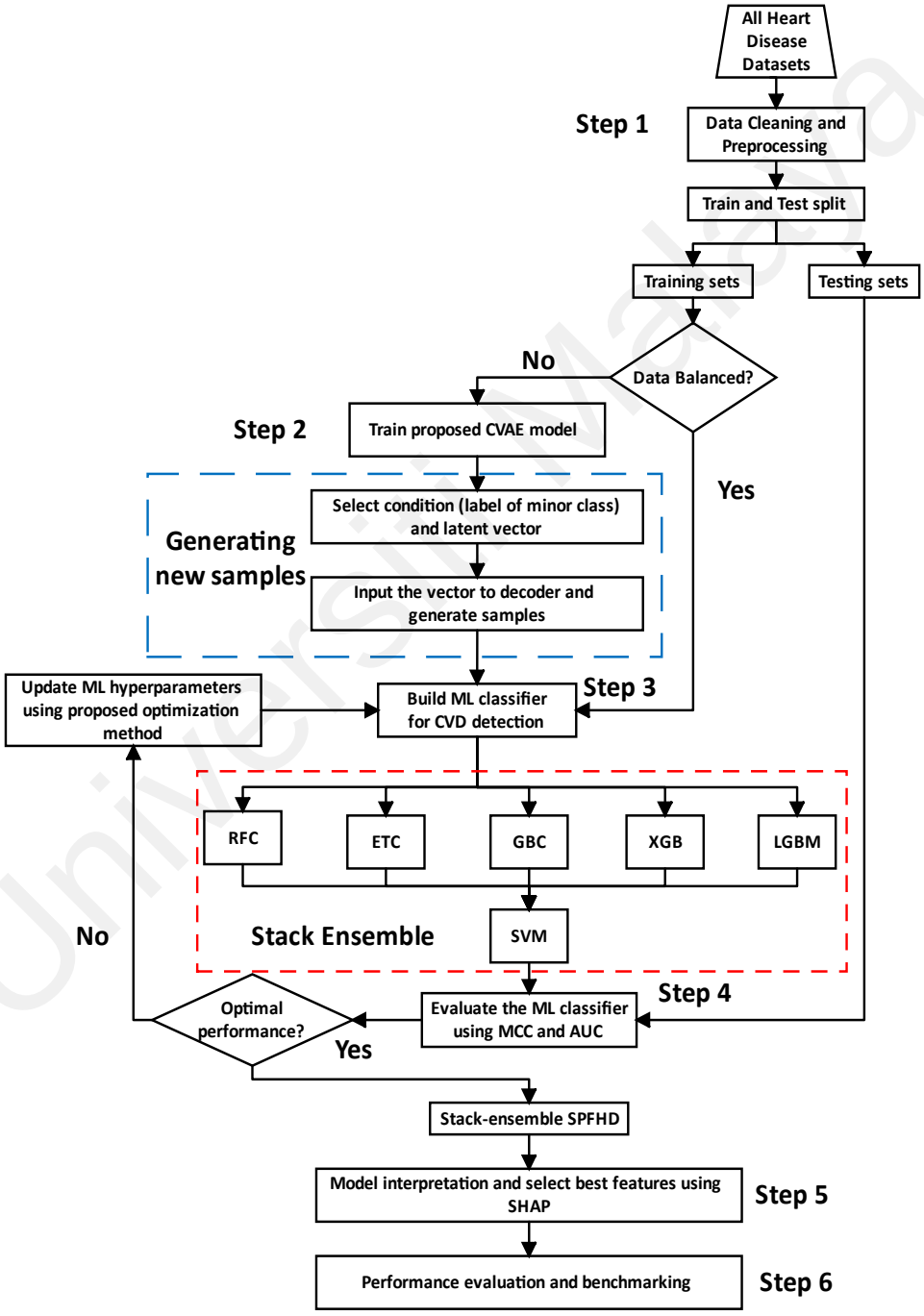


Figure 3.5: Flowchart of the proposed method two (SPFHD)

3.4.1 CVAE-based method for data balancing

Variational autoencoder (VAE) models (Kingma & Welling, 2013) are a variant of the classical autoencoder (AE) network (Bourlard & Kamp, 1988). Similar to an AE, a VAE comprises two coupled neural networks: an encoder and a decoder. The encoder is an inference model q (z|x) that maps input data x to a lower-dimensional latent variable space, z. In contrast, the decoder network receives the latent space z variables as input and outputs the probability distribution of the data p (x|z). A VAE differs from an AE network in forming a latent vector; rather than immediately producing a latent vector and maximizing the marginal log-likelihood, a vector of standard deviations (σ) and a vector of means (μ) are created and merged to form the latent vector. However, the network would not be able to learn the distribution that results from the encoder since the direct combination of these parameters in the latent vector z indicates that this is a continuous random variable. The reparameterization approach should be used to resolve this issue and describe the random variable z as a deterministic variable, with z depending on the parameters of the encoder output (μ; σ) and an additional variable epsilon sampled from a Gaussian distribution expressed in Eq. 3.10:

$$z^{(i,l)} = \mu^{(i)} + \sigma^{(i)} \odot \varepsilon^{(l)}, \varepsilon^{(l)} \sim \mathcal{N}(0, I) \quad 3.22$$

VAE further differ from classical AE in that they maximize the evidence lower bound (ELBO) on the marginal log-likelihood of p(x) presented in Eq. 3.10. The Kullback-Leibler (KL) divergence between the previous distribution p(z) and the encoder's distribution q(z|x) is referred to as KL(q(z|x)||p(z)) in Eq. 3.11. This term is a regularizer, assessing the information lost when p is represented by the distribution q. Fig. 3.3 depicts the comparative information flow of the two architectures (AE and VAE).

$$\min_p \mathbb{E}_{q(z|x)}[\log p(x|z)] - KL(q(z|x)||p(z)) \quad 3.23$$

CVAE is a variant of VAE (Sohn et al., 2015). In this type, conditional information y or other data information is introduced to the model in both the encoder and decoder. This addition makes the model conscious of the sample class that must be mapped into the encoder's latent space z, enhancing its capacity to distinguish between sample classes. As demonstrated in Eq. 3.12.

$$\min_{p} \mathbb{E}_{q(z|x,y)}[\log p(x|z,y)] - KL\,(q(z|x,y)||p(z|y)) \qquad 3.24$$

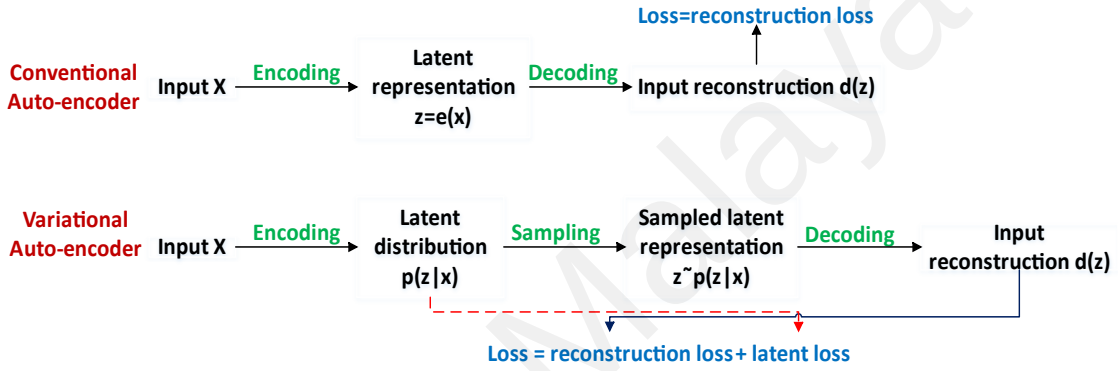


Figure 3.6: The difference between the information flow of a VAE and conventional AE

The illustration shown in Figure 3.6 depicts the information flow of the CVAE: the input (x and label y) of the encoder network, the intermediate latent space z produced by standard deviation (σ), mean (μ) vectors, and the distribution epsilon, and the network's output. Also depicted is the propagation of the loss throughout the model, derived in the objective function Eq. 3.12. This loss corresponds to an entire batch in a model iteration. It represents the variation that must be applied to both the encoder and the decoder.

The CVAE-based method is developed with encoder and decoder neural networks to balance the datasets by producing new samples for minority classes (Z-alizadeh and HD clinical datasets). Figure 3.7 illustrates the overall view of the proposed CVAE architecture used for data balancing. The proposed CVAE architecture is as follows. The encoder network contains three hidden layers holding 100, 50, and 25 neurons, while the

decoder network has a similar layout in reverse order, with three hidden layers comprising 25, 50, and 100 neurons. Also, the encoder network includes two outputs with eight neurons each, corresponding to latent space z size that serves as the decoder's input. Each hidden layer of the CVAE-based model employs ReLU activation functions, except for the latent variable layers and the decoder output layer, which employ linear and sigmoid activation functions, respectively. In addition, an extra input neuron is added to the encoder and decoder to incorporate label information.

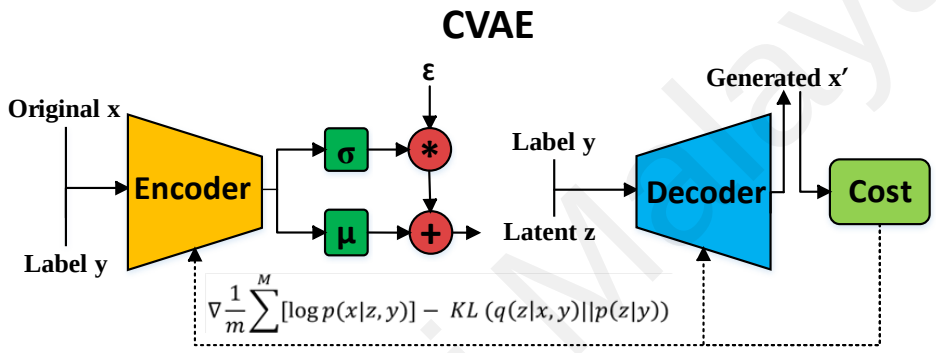


Figure 3.7: Schematic diagram of the functioning of the proposed CVAE model

To train the proposed CVAE, the loss function used consisted of a reconstruction term to make the encoding-decoding scheme effective. A regularization term to make the latent space regular, as shown in Equation 3.24. The mean squared error (MSE) is given as the first term. The KL divergence (Kingma et al., 2014), which calculates the difference between two distributions, is given as the second term. For each iteration, the proposed CVAE model was trained for ten epochs. At the beginning of each learning epoch, the input samples are randomly shuffled to ensure efficient network learning performance. The CVAE model is trained using Adam optimization with a learning rate of 0.01 throughout the experiment. Unless otherwise specified, the default settings of the open-source TensorFlow Framework were used to train the model. After training, the encoder network produces the covariances means, from which new latent vectors are sampled and used to create new samples by feeding them through the decoder; the sampling is

performed by varying the latent vectors z and selecting the condition (label of the minor class). The number of generated samples for each minor class is decided based on how many new instances are needed to balance the input dataset for training.

After the training phase of the proposed CVAE model, the encoder network yields covariance means, which act as the foundational structure for our sample generation. Leveraging this, we strategically sample new latent vectors 'z'. These vectors are infused with variability, achieved by sampling from a normal distribution, ensuring each vector possesses a unique profile. The generation of these vectors is paired with specific conditions corresponding to the labels of the minority class, directing the creation of samples representative of this class. New samples are produced by decoding these latent vectors alongside their associated conditions. The volume of samples generated is tailored to the minority class's needs. This methodology offers a nuanced, data-driven means to augment datasets, bringing them closer to balance while preserving the diversity and representativeness of the generated samples (Abdellatif et al., 2024) .

3.4.2 SPFHD Framework

To identify the presence and absence of HD, we apply a stacking method to develop SPFHD. Stacking is an ensemble learning strategy incorporating data from multiple prediction models to produce a stable stacked model. The technique uses a practical scheme to decrease the generalization error rate of multiple classifiers, as well as its stability and effectiveness, have been demonstrated (Charoenkwan et al., 2022). The stacking learning approach consists of two major stages, with predictive models in these two stages referred to as base and meta-learner, respectively. In the initial stage, base learners are applied, and their output is then integrated to reduce generalization errors.

In the first stage of stack learning, five tree-based ensemble learning classifiers have been applied to construct the first layer, including two bagging-based classifiers, RFC

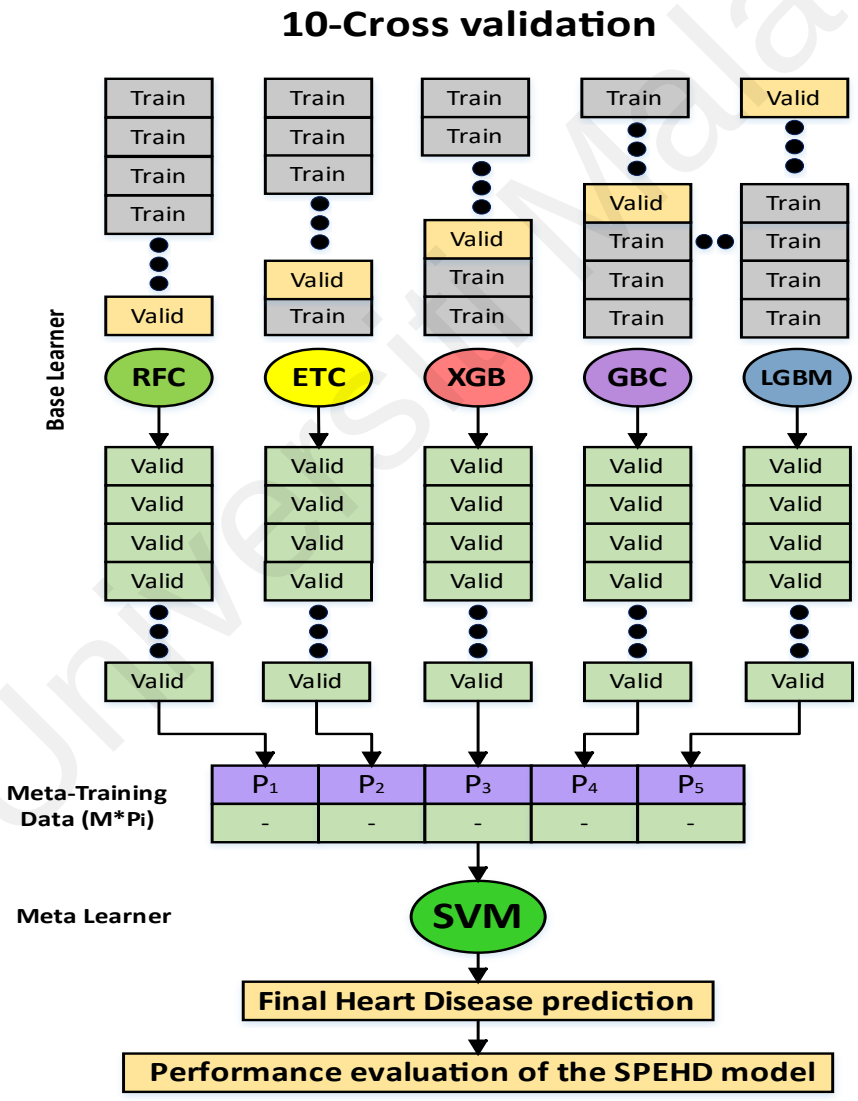
(Breiman, 2001) and ETC (Geurts et al., 2006). Also, three popular boosting-based classifiers gradient boosting classifier (GBC) (Friedman, 2001), XGB (Chen et al., 2015) and light gradient boosting machine (LGBM) (Ke et al., 2017). RFC is a bagging ensemble model whose individual model is a decision tree. The bagging method trains individual models in parallel, utilizing a random subset of the training data for each model. RFC is a well-known algorithm in computational biology and cardiovascular disease, and it is widely employed to solve a wide range of research concerns in cardiovascular illness (Ali et al., 2021; Gupta et al., 2019; Mohan et al., 2019). The sole difference between the RFC and ETC is the manner in which trees are constructed. Each decision tree in the additional tree classifier is constructed from the initial training sample. Random samples of *k* best features are used for decision making, and the Gini index is applied to determine the best feature for data partitioning in a tree. ETC has shown efficacy and consistency in numerous cardiac disease prediction tests (Ishaq et al., 2021). GBC is a boosting model that learns directly from the residual errors as opposed to updating the sample weight. GBC generates a new forecast by combining the past forecasts of all trained trees. GBC has two efficient extensions and implementations: XGB and LGBM. These techniques have been utilized successfully in computational biology since they efficiently deal with big datasets and parallel computing (Budholiya et al., 2020; Fitriyani et al., 2020; Kibria & Matin, 2022).

The output probability is computed using the outputs of the five base learners to train the meta-learner SVM model. In addition, the stacking approach is applied using the *k*-folds cv method. This approach extends the classic stacking algorithm by utilizing cv to arrange the input data for the meta-learner, hence avoiding overfitting (Aggarwal & Reddy, 2014). This approach divides the training dataset into *k*-folds (using ten folds). As k rounds, k-1 subsets are utilized to fit the base learners. In each iteration, the base learners subsequently validated on other fold that has not been utilized for model fitting. The

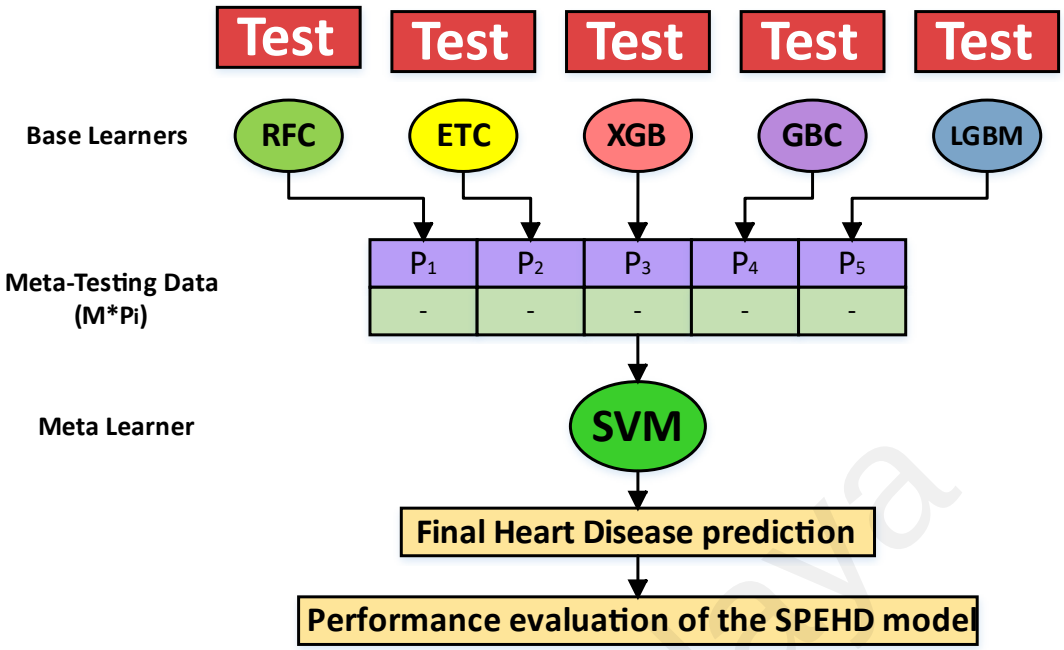
stacked predictions are then utilized as input for the meta-learner. Once training with the stacking cv process finishes, the base learners are fitted to the whole data set. The entire procedure for the SPFHD model during training and testing with the data partitioning is shown in Figure 3.8. To cut down the computational time for training and testing SPFHD, the base learners learn in parallel. Therefore, the computational time of the SPFHD can be computed as in Equation 3.25 instead of Equation 3.26.

$$SPFHDCT = \max(RFCCT + ETCCT + GBCCT + XGBCT + LGDMCT) + SVMCT \quad 3.25$$

$$SPFHDCT = RFCCT + ETCCT + GBCCT + XGBCT + LGDMCT + SVMCT \quad 3.26$$



(a)



(b)

Figure 3.8: The entire procedure for the SPFHD model during (a) training and (b) testing.

3.5 Performance evaluation metrics

Different performance metrics are considered to evaluate the proposed models' performance. This work utilized six performance metrics: accuracy, precision, recall, specificity, f1-score, and Mathew's Correlation Coefficient (MCC). The experiments are executed five times using different seeds to avoid randomness, and the mean value is considered. The six-evaluation metrics (accuracy, precision, recall, specificity, f1-score, and MCC, respectively) are summarized from Equations (3.27-3.32) as follows:

$$\text{Accuracy (ACC)} = \frac{TP + TN}{TP + FN + FP + TN} \quad 3.27$$

$$Precision\ (PPV) = \frac{TP}{TP + FP} \quad 3.28$$

$$Recall\ (TPR) = \frac{TP}{TP + FN} \tag{3.29}$$

$$\text{Specificity (SPC)} = \frac{TN}{TN + FP} \tag{3.30}$$

$$F1\ = \frac{2 * percision * recall}{percision + recall} \tag{3.31}$$

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{3.32}$$

3.6 Model interpretation using SHapley Additive exPlanations (SHAP)

Understanding why ML models make specific predictions can be significant for investigating the uncertainty in heart disease prediction. SHAP is a strategy suggested by Lundberg and Lee to explain the individual predictions of ML models based on the game theory approach Shapely (Lundberg & Lee, 2017). The framework integrated the previously proposed methods for interpretation of ML models (LIME, DeepLIFT, Shapley sampling values, Shapley regression values, Layer-Wise Relevance Propagation, and tree interpreter). The SHAP framework utilizes cooperative game theory by assigning a relevance score to each attribute based on its impact on the model forecast. SHAP creates a simple descriptive model to describe the prediction *d* using the formula $X_i'\{0,1\}^M$ where *i* is the instance and M is the number of features. The local explanation model is represented by Eq. 3.19.

$$\tau(\dot{X}) = \delta_o + \sum_{i=1}^{M} \delta_i \dot{X}_i \tag{3.19}$$

The SHAP values assigned a significant value γ_i to each feature, signifying the effect of training the model using that feature. The SHAP values can be estimated as follows in the cooperative game theory, as shown in Eq. 3.20.

$$\gamma_i = \sum_{C \subseteq D \setminus \{i\}} \frac{|C|!\,(|D| - |C| - 1)!}{|D|!} \left[d_{C \cup \{i\}}\left(X_{C \cup \{i\}}\right) - d_C(X_C) \right] \quad 3.20$$

Where D represents the feature set, and C represents the feature subset which removes the i^{th} feature in D. Then, two models, $d_{C \cup \{i\}}$ and f_S are retrained. Finally, predictions from these two models are compared to the current input $d_{C \cup \{i\}}\left(X_{C \cup \{i\}}\right) - d_C(X_C)$, where X_C represents the values of the input features in the feature subset C. Furthermore, the SHAP technique offers an exciting capability to generate interpretable predictions and assign each feature weight for the forecasts of the complex ensemble models (Mubarak et al., 2022). In the context of applying SHAP to the SPFHD to have a better understanding of how SPFHD is making predictions, the following steps sum it up:

1. Stacked Model Overview: A SPFHD model combines predictions from five tree-based ML algorithms to create a more accurate ensemble prediction. Each of these individual algorithms could have its unique way of processing features, and therefore, understanding the cumulative importance of a feature across these models is crucial.

2. Data Input: Feed the raw input data to the SPFHD model. Each base model in the stack will generate its prediction or an intermediate representation of the data.

3. Compute SHAP Values: For each base model in the SPFHD:

- Calculate the SHAP values for each feature. This gives you an insight into how each feature influences the prediction for that specific base model.
- The SHAP value for a feature signifies how much the prediction deviates from the baseline (usually the average prediction) when that feature is considered.

4. Aggregating SHAP Values: Given the layered nature of a stacked model, multiple sets of SHAP values corresponding to each base model:

- **Weighted Aggregation**: Aggregate the SHAP values depending on the weight or influence of each base model on the final prediction. Since all have the same weight, the average of SHAP values is taken.
- The aggregated SHAP values will represent the overall influence of each feature across the entire stacked model.

5. Analyzing Feature Importance:

- **Summary Plot**: Visualize the aggregated SHAP values using a summary plot. Features will be ranked based on their importance, with the magnitude and direction of the SHAP value indicating how much and in which direction they influence the final prediction.
- **Positive vs. Negative Influence**: Features with positive SHAP values push the model's output higher (than the baseline), while those with negative SHAP values push the output lower. The magnitude gives an idea of the strength of this influence. To understand this more, here is an example: Positive SHAP Value for "Age": When a particular sample has a positive SHAP value for the "Age" feature, it signifies that the age of this sample contributes to an increased likelihood of the model predicting heart disease. This means that for this specific instance, the model views the age in question as increasing the risk of heart disease. Negative SHAP Value for "Age": On the other hand, if a sample has a negative SHAP value for the "Age" feature, it means the age of this sample is leading the model to predict a decreased likelihood of heart disease. In this instance, the model perceives the given age as reducing the risk of heart disease.

As a conclusion of interpreting the positive and negative SHAP values for "Age", If we observe that the positive SHAP values for the "Age" feature predominantly occur for samples with higher ages, it reinforces the idea that older age contributes to a higher

predicted risk of heart disease. Similarly, suppose negative SHAP values for the "Age" feature are commonly seen for samples with younger ages. In that case, it supports the idea that the model sees younger age have less risk of having heart disease.

6. Interpreting the Results:

- Features with larger absolute SHAP values have a stronger impact on the model's prediction.
- The sign of the SHAP value indicates whether the presence of a feature increases or decreases the prediction.
- Features closer to the top of the summary plot are more influential in the stacked model's decision-making.

Consequently, we utilized the SHAP approach to understand the SPFHD results. The SHAP method's use in this work can be illustrated in Figure 3.9. The entire process of each step in the flowchart will be elaborated in Figure 3.5 as follows:

Step 1: Data collection, preprocessing, and partitioning. This process is detailed in Data Collection and Preprocessing 3.2; four datasets are used to validate the proposed model, including Cleveland, Statlog, Z-Alizadeh Sani, and HD clinical records. This step embraces four stages: eliminating samples with missing attributes, label grouping, data normalizing using mean and standard deviation, and dataset splitting.

Step 2: the proposed CVAE model is trained, and new samples from the minor class are generated for data balancing. Otherwise, if the data is balanced, it directly goes to the next step.

Step 3: The stack ensemble model is developed after the data balancing. Where the model is formed of one layer containing five base learners (RFC, ETR, GBC, XGB, and LGBM), the base learners' prediction will be combined as an input for the meta-learner.

Step 4: The evaluation of the ML classifiers will be conducted in this step by utilizing MCC and F1. When the model fulfills the optimal performance conditions, it moves to the next step. Otherwise, it will return to update the model HPs.

Step 5: In this step, the model interpretation is handled for a deeper insight into the learning mechanism of the SPFHD and a better understanding of features utilizing the SHAP test. The most contributed features will be selected.

Step 6: Finally, the performance evaluation and benchmarking will be conducted in this step.

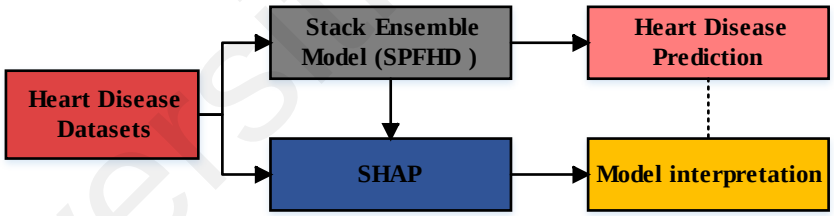


Figure 3.9: The implementation of the SHAP framework in interpreting the SPFHD model.

3.7 Summary

This chapter describes the two methodologies used to accomplish the research objectives. The CVD datasets and the data gathering are detailed in detail. The data processing is given for classification methods. In addition, the proposed approaches (IWRF for data balancing on the algorithm level and CVAE for data balancing on the data level) for detecting CVD are expanded independently. The first methodology detects the CVD and solves the problem of the unbalanced data on the algorithm level by

assigning a weight for each class with a larger weight for the minor class using IWRF. In contrast, the second proposed methodology tackles the imbalanced problem on the data level by balancing the data distribution by generating new samples from a minor class by employing CVAE. In addition, the methodology of alternative detection approaches, such as single and hybrid models [RFC, ETC, XGB, and LGBM], data balancing strategies combined with other classifiers, such as SMOTE-RFC, are compared to the methodology proposed. Performance metrics are described to evaluate the accuracy of detection systems by comparing anticipated and actual results. In addition, the approach of various optimization algorithms, namely GA, PSO, RS, Hyperband, and BO, is described to tune the hyperparameters of the proposed methods in order to improve CVD detection accuracy.

CHAPTER 4: RESULT AND DISCUSSION

This section presents the proposed models' results for different data sets according to the performance metrics. Further, the feature selection results, classification results, and SHAP results are demonstrated in each subsection. Finally, a comparative study is elaborated at the end of this section to highlight the effectiveness of the proposed models.

4.1 The proposed Inf-FSs-IWRF model.

This part proposes an effective method to predict CVD and patients' survival: Inf-FS$_s$ to rank the features by importance and select the best features, IWRF to predict CVD, and BO to find the best weighting coefficient. In addition, two public datasets were chosen to develop the model and test the model, the Statlog dataset (*Statlog (Heart) Data Set*) to detect the absence and presence of CVD and the heart failure clinical record dataset (Ahmad et al., 2017) to predict the patients' survival. Hence, an ML algorithm is developed to diagnose CVD and patient survival to assist healthcare professionals. As a result, early treatment might be implemented to avoid the deaths caused by late CVD detection. Moreover, this section presents the feature selection results first, followed by HD presence and survival classification performed for both datasets. The developed model was built and tested for HD on Statlog and heart failure clinical record datasets. The Statlog dataset consists of 14 attributes with the status label, 270 cases, 150 for HD absence and 120 for HD presence. The heart failure clinical record dataset consists of 13 attributes with the survival label, 299 total cases, 203 patients survive, and 96 patients deceased. We used a 10-fold cv procedure in our experiment to avoid overfitting (Kohavi, 1995). The model performance is evaluated using six performance metrics.

4.2 Feature Selection Results for the proposed Inf-FSs-IWRF model.

Inf-FS$_s$-based feature selection is conducted at each stage of the 10-fold cv utilizing the training data. The Inf-FS$_s$ method ranks and weights each feature in the 13 and 12

features pool for both datasets. Table 4.1 summarizes the features and associated Inf-FS$_s$ weights for a given fold. The top ten attributes for each validation fold are chosen from these ranking features automatically. Nine characteristics appear in both datasets' top ten features for each of the 10-folds of the training data evaluated. As a result, the presence and survival classifications use these nine features. The selected features for both datasets are listed in Table 4.2.

A comparison of feature rankings and their associated importance weights as determined by the Inf-FSs method, across two datasets: the Statlog dataset and the heart disease clinical record is presented in Table 4.1. In the Statlog dataset, a total of 13 features are listed, ranging from 'Age' to 'Thal'. 'Thal' emerges as the most critical feature, occupying the top rank with the highest weight of 13.165, while 'Chol' stands at the bottom of the list at rank 13, holding the lowest weight of 5.619. Features like 'Gender' and 'Exang' also feature prominently, securing the third and second ranks with weights of 9.928 and 11.092, respectively. Other attributes fall in between, each with a distinct rank and weight that reflects its importance according to the Inf-FSs method.

In parallel, the heart disease clinical record dataset encompasses 12 features with 'Age' and 'Time' bookending the list. 'Anemia' is deemed the most important feature with the highest weight of 11.191 and a corresponding rank of 1. In contrast, 'CPK' is identified as the least significant feature with a weight of 6.403, ranking at 11. Other features such as 'Diabetes' and 'High BP' are also highlighted for their significance, holding the second and third ranks with weights of 11.135 and 10.768, respectively. The weights and ranks collectively represent the features' relative importance in this dataset for the given fold of cross-validation.

Table 4.1: Feature ranking and weight importance determined by Inf-FS$_s$

Statlog dataset			Heart disease clinical record		
Attributes	Rank	Weight	Attributes	Rank	Weight
Age	11	6.133	Age	7	7.203
Gender	3	9.928	Anaemia	1	11.191
CP	6	8.898	CPK	11	6.403
Tresthps	12	5.793	Diabetes	2	11.135
Chol	13	5.619	Ejection_fraction	8	6.849
Fbs	10	7.159	High BP	3	10.768
Restecg	4	9.875	Platelets	12	6.382
Thalach	8	7.758	Serum_creatinine	10	6.404
Exang	2	11.092	Serum_sodium	9	6.728
Oldpeak	9	7.722	Gender	4	10.766
Slope	7	7.897	Smoking	5	10.587
Ca	5	9.231	Time	6	8.04
Thal	1	13.165			

Building on the ranking and weighting of features, Table 4.2 combines the selected features identified as the most significant for each dataset post the Inf-FSs feature selection process. For the Statlog dataset, the selection narrows down to 'Thal', 'Exang', 'Gender', 'Restecg', 'Ca', 'CP', 'Slope', 'Thalach', and 'Oldpeak'. These features are pinpointed as the most relevant for the Statlog dataset's predictive modeling. Similarly, for the heart disease clinical record, the method selects 'Anemia', 'Diabetes', 'High BP',

'Gender', 'Smoking', 'Time', 'Age', 'Ejection_fraction', and 'Serum_sodium'. These features are distinguished as the most predictive for outcomes within the heart disease clinical records, implying their strong potential for influencing the model's performance. Both tables together provide a clear depiction of the outcome of applying Inf-FSs within a ten-fold cross-validation framework, underscoring the features that consistently hold the most predictive power across multiple folds.

Table 4.2: Selected features of both datasets.

Dataset	Selected features
Statlog	Thal, Exang, Gender, Restecg, Ca, CP, Slope, Thalach, Oldpeak
HD clinical record	Anemia, Diabetes, High BP, Gender, Smoking, Time, Age, Ejection_fraction, Serum_sodium

4.2.1 Classification results for the proposed Inf-FSs-IWRF model.

The developed IWRF model was used for both datasets and showed significant improvement in prediction accuracy compared to existing models. For comparison, we chose six distinct machine learning models (G-NB, LR, SVM, kNN, XGBoost, and RF) frequently utilized in the research field and have a proven record of accuracy and efficiency. The results of different ML models are presented in Table 4.3 and Table 4.4 for Statlog and HD clinical records, respectively, including the effects of both with and without FS. IWRF performed better across both datasets than other ML models achieving accuracy, F-measure, and MCC up to 95.5%, 94%, and 0.9 for Statlog, 93.3%, 86%, and 0.81, for the HD clinical dataset, respectively. Also, it is noted that all models have been improved when using FS on both datasets, especially for IWRF, by reaching accuracy, F-measure, and MCC up to 97.7%, 97%, and 0.95 for Statlog, and 95.9%, 91.3%, and 0.88, for HD clinical dataset, respectively.

Furthermore, it can be shown from the results that IWRF achieved better results than the standard RF model in handling the imbalanced data, where IWRF improved the performance for detecting CVD and patients' survival by 3.7% and 5%, respectively, after FS. Recognizing the minority class sufficiently during classification is difficult because the standard RF and the other models used to learn from data input are biased towards the majority class. With the benefit of feature selection, doctors can forecast the survival of patients and the presence of HD by assessing the essential attributes.

To get another point, the IWRF was compared with SMOTE as it is commonly used in handling unbalanced datasets. As with any sampling technique, SMOTE is not a stand-alone classifier but can be integrated with any classifier. For a fair comparison, SMOTE was combined with RF and then compared with IWRF. Table 4.5 and Table 4.6 present the results of IWRF against base RF with SMOTE for both datasets. Moreover, we employed BO for tuning SMOTE hyperparameters (sampling ratio and k-neighbors) and (α, p) for IWRF, while the other hyperparameters such n_estimators, max_depth, max_features, and min_samples_split, are set for the default values as in Sklearn library. The findings showed that IWRF achieved higher results than base RF with SMOTE since SMOTE has several drawbacks related to overlap and noisy information. It regularly assigns a global k-neighbor but ignores the local distribution features (Cheng et al., 2019; Sáez et al., 2015). The hyperparameter tuning improved model prediction accuracy, but it showed more impact on SMOTE. Increasing the k-neighbor value to compensate for the imbalance ratio may be effective in SMOTE. The results illustrated in Figure 4.1 show that the improvement achieved by the proposed IWRF is higher than SMOTE-RF compared to the base RF classifier. The proposed model improved the performance of CVD detection by 3.62%, 4.82%, for the Statlog dataset, and 6.3%, 11.98% for HD clinical records in terms of accuracy and f-measure, respectively.

Table 4.3: Performance evaluation of Statlog dataset for the proposed FS method

Model	Without FS						With FS					
	Acc.	Pre.	Recall	F-measure	SPC	MCC	Acc.	Pre.	Recall	F-measure	SPC	MCC
SVC	0.921	0.947	0.847	0.894	0.969	0.836	0.929	0.948	0.866	0.905	0.969	0.852
kNN	0.87	0.886	0.771	0.821	0.933	0.728	0.895	0.914	0.809	0.857	0.951	0.781
G-NB	0.907	0.944	0.81	0.872	0.97	0.806	0.907	0.9	0.857	0.878	0.939	0.804
LR	0.903	0.916	0.828	0.87	0.951	0.797	0.907	0.917	0.838	0.875	0.951	0.804
XGBoost	0.933	0.939	0.885	0.91	0.963	0.859	0.944	0.949	0.904	0.926	0.969	0.882
RF	0.929	0.938	0.876	0.905	0.963	0.851	0.94	0.949	0.895	0.92	0.969	0.875
IWRF	0.955	0.98	0.904	0.94	0.987	0.906	**0.977**	**0.963**	**0.98**	**0.97**	**0.975**	**0.954**

Table 4.4: Performance evaluation of HD clinical records dataset for the proposed FS method

Model	Without FS						With FS					
	Acc.	Pre.	Recall	F-measure	SPC	MCC	Acc.	Pre.	Recall	F-measure	SPC	MCC
SVC	0.839	0.762	0.677	0.716	0.909	0.609	0.849	0.771	0.711	0.74	0.909	0.636
kNN	0.786	0.699	0.511	0.576	0.904	0.459	0.809	0.712	0.622	0.663	0.89	0.535
G-NB	0.833	0.75	0.667	0.706	0.905	0.592	0.85	0.765	0.722	0.743	0.9	0.63
LR	0.843	0.759	0.7	0.728	0.905	0.619	0.839	0.756	0.689	0.72	0.9	0.61
XGBoost	0.889	0.804	0.801	0.802	0.926	0.726	0.912	0.827	0.83	0.827	0.942	0.769
RF	0.893	0.813	0.801	0.806	0.93	0.733	0.909	0.817	0.83	0.823	0.937	0.762
IWRF	0.933	0.851	0.871	0.86	0.952	0.814	**0.959**	**0.926**	**0.9**	**0.913**	**0.978**	**0.881**

Table 4.5: Comparison results between IWRF and SMOTE-RF on Statlog dataset

Model	Without Optimization						With Optimization					
	Acc.	Pre.	Recall	F-measure	SPC	MCC	Acc.	Pre.	Recall	F-measure	SPC	MCC
SMOTE-RF	0.947	0.952	0.912	0.93	0.97	0.891	0.978	0.979	0.965	0.972	0.986	0.955
IWRF	0.977	0.963	0.98	0.97	0.975	0.954	**0.983**	**0.986**	**0.972**	**0.979**	**0.991**	**0.966**

Table 4.6: Comparison results between IWRF and SMOTE-RF on HD clinical record dataset

Model	Without Optimization						With Optimization					
	Acc.	Pre.	Recall	F-measure	SPC	MCC	Acc.	Pre.	Recall	F-measure	SPC	MCC
SMOTE-RF	0.939	0.866	0.883	0.872	0.956	0.831	0.962	0.922	0.919	0.919	0.975	0.892
IWRF	0.959	0.926	0.9	0.913	0.978	0.881	**0.972**	**0.944**	**0.943**	**0.943**	**0.982**	**0.922**

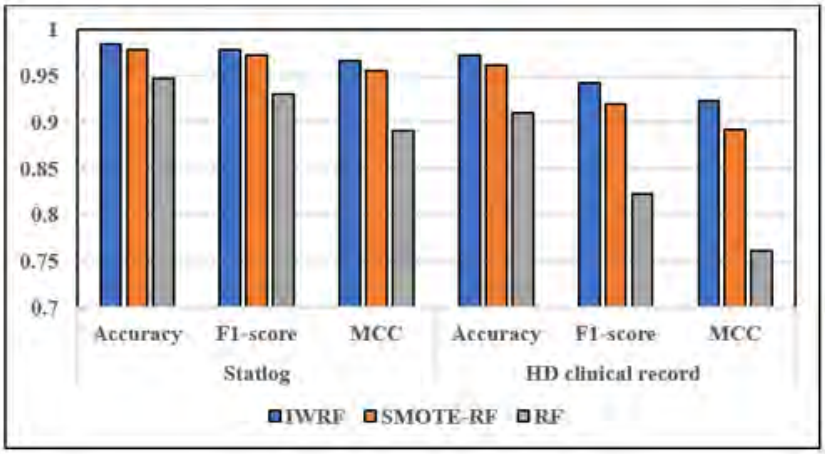


Figure 4.1: The comparison between the proposed IWRF and SMOTE-RF.

4.3 The proposed Stack Predictor for Heart Disease (SPFHD) model.

This section presents the classification results between the proposed method (SPFHD) and five base learners. In addition, the model interpretation using the SHAP framework for deeper insight and feature selection is demonstrated. Finally, a comparison between the proposed model with the previous works form the literature is conducted to highlight the supremacy of the proposed model.

4.3.1 Classification results for the proposed SPFHD model

The performance of the SPFHD with five different base learners is compared on the testing set for the four different datasets using the performance metrics to evaluate the efficacy of this strategy and the extent to which the stacked SPFHD model learns from the base learners. The results in Table 4.7 show the comparison between the developed stack ensemble and the five base learners using the default hyperparameters on the Cleveland and Statlog datasets. The results indicated that the stacked SPFHD model performed the best in terms of Acc, F1, and MCC for predicting HD. Furthermore, in terms of six performance parameters, SPFHD had the best performance for predicting the existence of HD on the Cleveland and Statlog datasets. In the case of Cleveland, the SPFHD achieved a result of 95.2, 93.42, 94.67, 95.56, and 0.9 for ACC, PPV, TPR, F1,

and MCC, respectively. On the other hand, in the Statlog dataset, the XGB outperformed SPFHD only in the TPR with a value of 93.33.

Table 4.7: The results for the SPFHD and the base models on the balanced datasets using the default hyperparameters.

Model	Cleveland					
	ACC	PPV	TPR	F1	SPC	MCC
RFC	92.00	91.19	88.66	89.82	94.22	0.83
ETC	93.33	92.54	90.67	91.56	95.11	0.86
GBC	92.53	91.81	89.33	90.53	94.67	0.84
XGB	94.93	93.37	94.00	93.65	**95.56**	0.89
LGBM	94.40	93.28	92.67	92.95	**95.56**	0.88
SPFHD	**95.20**	**93.42**	**94.67**	**94.03**	**95.56**	**0.90**
Model	Satalog					
	ACC	PPV	TPR	F1	SPC	MCC
RFC	93.91	93.95	90.48	91.89	96.10	0.87
ETC	94.44	96.36	89.52	92.51	**97.58**	0.89
GBC	93.91	94.07	90.48	91.92	96.10	0.87
XGB	94.44	92.60	**93.33**	92.80	95.09	0.88
LGBM	94.81	95.39	91.43	93.10	96.97	0.89
SPFHD	**95.18**	**96.48**	91.43	**93.50**	97.52	**0.90**

In this part, the effectiveness of a CVAE in addressing data imbalances is discussed. Other classical data imbalance handling methods, such as ADASYN and SMOTE, are applied to generate balanced datasets, which are then classified using the proposed SPFHD and other base learners. Additionally, no data balance handle is performed. Each

experiment is repeated five times for a more robust and reliable analysis, and the findings are averaged. The results in Table 4.8 compares the test set's performance with the Acc, PPV, TPR, F1, SPC, MCC, and Gmean values highlighted in bold of the two unbalanced datasets (Z-alizadeh and HDclinical records).

Table 4.8: The results for the SPFHD with different data balancing methods on two datasets

Model	Z-alizadeh						
	Acc	PPV	TPR	F1	SPC	MCC	G-mean
CVAE-SPFHD	**96.55**	**94.93**	98.64	**96.67**	**94.42**	**0.93**	**96.46**
Adasyn-SPFHD	95.63	92.78	**99.09**	95.83	92.09	0.91	95.53
SMOTE-SPFHD	96.09	93.58	**99.09**	96.26	93.02	0.92	96.01
Original	87.48	90.61	93.00	91.78	70.67	0.65	81.07
Model	HD clinical records						
	Acc	PPV	TPR	F1	SPC	MCC	G-mean
CVAE-SPFHD	**94.33**	**97.50**	**88.80**	**92.85**	**98.28**	**0.88**	**93.38**
Adasyn-SPFHD	93.33	95.93	86.11	90.68	97.89	0.85	91.77
SMOTE-SPFHD	93.33	96.39	87.20	91.56	97.71	0.86	92.28
Original	90.00	89.32	77.14	82.75	95.80	0.76	85.96

According to Table 4.8, the combination of CVAE-SPFHD provides the best Acc, SPC, and G-mean performance on both the Z-alizadeh and HD clinical record test sets. At the same time, SMOTE and Adasyn with SPFHD achieved higher TPR only on the Z-alizadeh test. The CVAE-SPFHD model obtained the best G-mean for the Z-alizadeh dataset (96.46), which is higher than the second-best technique (SMOTE-SPFHD, with

G-mean = 96.01). The proposed CVAE-SPFHD (G-mean = 93.38) strategy for the HD clinical records dataset is also better than the best approach SMOTE-SPFHD with G-mean = 92.28.

Table 4.9: The results for the SPFHD and base learners with CVAE-based method for data balancing methods on two datasets

Model	Z-alizadeh						
	Acc	PPV	TPR	F1	SPC	MCC	G-mean
RFC	93.33	92.48	94.55	93.43	92.09	0.87	93.27
ETC	93.79	91.79	96.36	93.99	91.16	0.88	93.71
GBC	92.64	91.61	94.09	92.81	91.16	0.85	92.60
XGB	95.40	93.11	98.18	95.57	92.56	0.91	95.32
LGBM	95.17	93.06	97.73	95.32	92.56	0.90	95.10
SPFHD	**96.55**	**94.93**	**98.64**	**96.67**	**94.42**	**0.93**	**96.46**
Model	HD clinical records						
	Acc	PPV	TPR	F1	SPC	MCC	G-mean
RFC	93.00	96.77	86.40	91.16	97.71	0.85	91.84
ETC	92.78	96.67	83.80	89.71	98.46	0.84	90.79
GBC	92.22	92.23	86.91	89.41	95.53	0.83	91.08
XGB	93.33	92.82	88.60	90.66	96.10	0.85	92.24
LGBM	93.00	95.42	87.20	91.17	97.14	0.85	92.01
SPFHD	**94.33**	**97.50**	**88.80**	**92.85**	**98.28**	**0.88**	**93.38**

Furthermore, the effectiveness of the proposed SPFHD is compared to that of the base learner algorithms with the CVAE-based method. Table 4.9 highlights the average scores for the test set performance for the Z-alizadeh and HD clinical records datasets. With Acc

values of 96.55 and 94.33, MCC values of 0.93 and 0.88, and G-mean values of 96.46 and 93.38 on Z-alizadeh and HD clinical records test sets, respectively, the proposed method (SPFHD) method surpasses the base learner algorithms for predicting HD. In addition, the proposed technique increased specificity by 23.75 % for Z-alizadeh dataset and sensitivity (TPR) by 11.66 % for the HD clinical records dataset.

4.3.2 Hyperparameter optimization results for the proposed SPFHD model

Three HPO methods (BO, PSO, and GA) are employed to reduce the effort of ML hyperparameter tuning, especially for complex ML models with many HPs. The proposed model and the base learners with their default HP configuration are trained and evaluated as baseline models in the first step. Then, each HPO algorithm is used for the ML models in order to evaluate and compare their classification performance. It is evident from Table 4.7 that using the default HP settings does not produce the best model performance throughout the experiments across four datasets, highlighting the significance of employing HPO techniques. Table 4.10 shows the results after applying HPO methods to the SPFHD. The performance of SPFHD improves sharply; for example, the MCC increases from 0.9 to 0.98, 0.97, and 0.97 after applying BO, PSO, and GA, respectively, for the Cleveland dataset. It is clear from the results that BO achieved the highest improvement rate across the four datasets. Also, GA achieved the lowest improvement rate across the datasets (except for the HD clinical records dataset, PSO achieved a lower enhancement.

Table 4.10: The results for the SPFHD with different HPO methods on different datasets

Model	Cleveland					
	ACC	PPV	TPR	F1	SPC	MCC
SPFHD	95.20	93.42	94.67	94.03	95.56	0.90
GA-SPFHD	98.67	97.46	**99.33**	98.36	98.22	0.97
PSO-SPFHD	98.67	98.06	98.67	98.35	**98.67**	0.97
BO-SPFHD	**98.93**	**98.10**	**99.33**	**98.69**	**98.67**	**0.98**
Model	**Statlog**					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	95.18	96.48	91.43	93.50	97.52	0.90
GA-SPFHD	98.41	97.43	**98.64**	97.99	98.23	0.97
PSO-SPFHD	98.94	98.76	**98.64**	98.65	99.13	0.98
BO-SPFHD	**99.02**	**99.24**	98.25	**98.72**	**99.48**	**0.98**
Model	**Z-Alizadeh Sani**					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	96.55	94.93	98.64	96.67	94.42	0.93
GA-SPFHD	97.93	98.26	97.73	97.95	98.14	0.96
PSO-SPFHD	98.62	**98.69**	98.64	98.64	98.60	0.97
BO-SPFHD	**98.85**	**98.69**	**99.09**	**98.87**	**98.60**	**0.98**
Model	**HD clinical records**					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	94.33	97.50	88.80	92.85	98.28	0.88
GA-SPFHD	97.77	97.99	97.44	97.69	98.14	0.96
PSO-SPFHD	97.53	97.97	96.92	97.42	98.14	0.95
BO-SPFHD	**98.49**	**98.99**	**97.95**	**98.45**	**99.09**	**0.97**

4.3.3 Model interpretation for the proposed SPFHD model

Here, the mechanisms related to SPFHD's predicted reliability are assessed. Five tree-based ensemble algorithms (RFC, ETC, GBC, XGB, and LGBM) are employed to develop the basis classifiers of SPFHD. For the predictions of the complex ensemble models, the SHapley Additive exPlanations (SHAP) technique provides an attractive ability to generate interpretable predictions and assign a relevance score to each feature. Consequently, the SHAP approach is used to understand the stacked SPFHD findings. SHAP is utilized to determine the essential SPFHD attributes. Figure 4.2 displays the features based on SHAP values for heart disease presence and failure ordered by the sum of SHAP-value magnitudes across all observations and displays the distribution of each feature's impact on SPFHD output. The majority of variables with high values corresponded to positive SHAP values, whereas those with low values corresponded to negative SHAP values, according to the findings. Calculating the risk stratification is based on the risk factors dependent on age and sex. First, age alone with no comorbidities plays a risk in increasing the CVD events from 2% at age 40-50 years to 32.5 % at the age of 100 (Savji et al., 2013). Second, the male sex is at a slightly higher risk of developing CVD than females due to unknown mechanisms (Kappert et al., 2012; Tunstall-Pedoe et al., 1999). Now for the other risk factors:

1. Patients having increased BP (at rest or having HTN) are at greater risk of developing CVD (Franklin et al., 2001) at all ages, but it is more considerable after the age of 55 years for males and 65 years for females (James et al., 2014).

2. According to the INTERHEART study, DM plays a risk in increasing CVD, while it doesn't play a major role in hospitalization or mortality due to CVD.

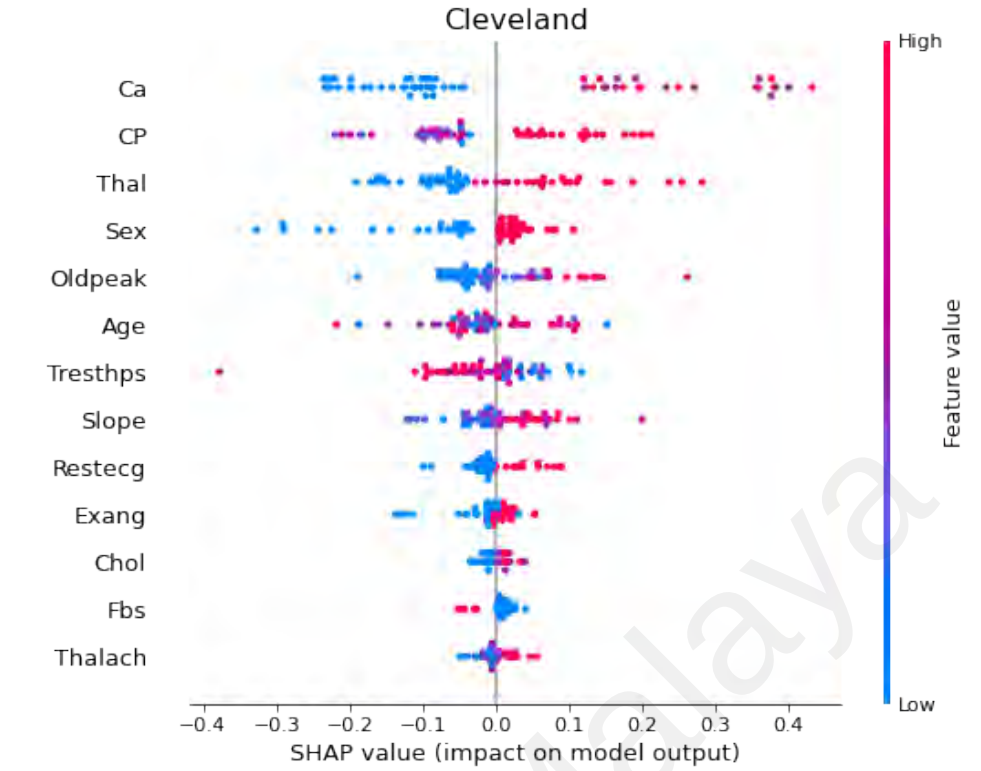
3. CKD increase the risk of CVD, according to the American Heart Association (AHA) and American College of Cardiology (ACC).

4. Smoking, according to the INTERHEART study, increases the risk of CVD.
5. Also, increased BMI increases the risk of CVD (Harris et al., 1988).
6. Family history independently increases the risk of CVD, mainly in younger adults.
7. Increased LDL, total cholesterol, and decreased HDL play a risk in developing CVD but to a lesser extent, same for FBS (Eckel et al., 2004; Tirosh et al., 2011).
8. But before listing the risk factors, you must consider any CV event by doing the following investigation:
    a. History taking for any event, chest pain (typical or atypical), and dyspnea.
    b. Physical examination starts with vitals focusing on resting blood pressure and chest examination, including heart sounds and murmurs.
    c. Labs, including electrolytes and CPK.
    d. ECG (resting as a baseline mainly in previous admissions to the ER or hospitalization to compare it with the new ones).
    e. Echography to calculate the EF.
    f. Any other investigation

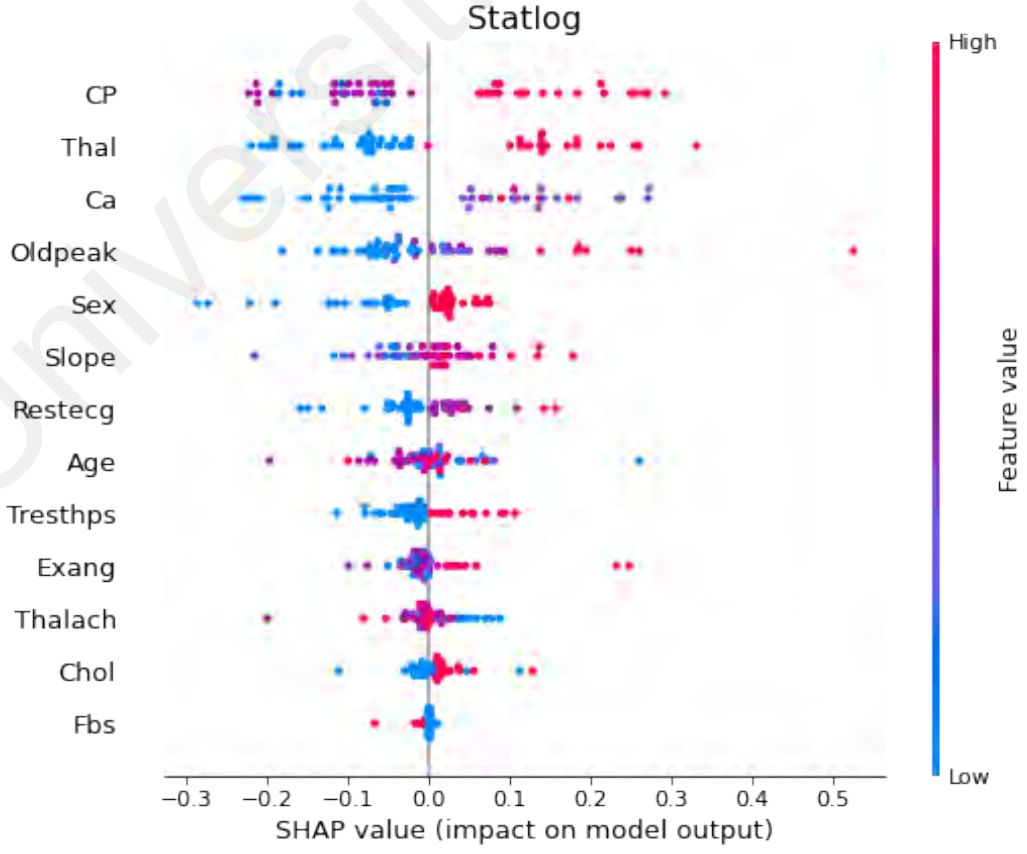
After that, the risk stratification and the mortality rate can be calculated. Going back to diagnosing and extracting the prognosis of the CVD, there are multiple factors -taken by the internist- that affect this pathway, mostly the age, sex, ECG, HTN, symptoms and signs in the physical examination, which is also found to be in the datasets such as Statlog and Cleveland datasets to assist the physician for taking the right decision for example as shown in Fig. 4.3 (a, b) as the age increases (changes from blue to red), the risk of having CVD increases also, and any change in the resting ECG may indicate further investigation for the diagnosis of CVD (as the indicator changes from blue to red the need of further investigation increases) and the same for the other parameters. At the same time, the Z

dataset specifies each component and adds much more details to the features in the Statlog and Cleveland datasets and has an add on features like TTE-EF, where when it increases, the outcome is more favorable (as the indicator changes from red to blue the result is more favorable). On the other hand, in DM, when the patient has the disease, the outcome is less favorable. The prognosis of CVD is poor (as the indicator changes from blue to red, CVD is worst, and as fact, Type 2 DM is associated with poor prognosis (Al-Delaimy et al., 2004; Almdal et al., 2004; Kannel & McGee, 1979), which makes it more specific for the diagnosis of CVD and a better indicator for the prognosis as shown in fig. 4.3 (c).

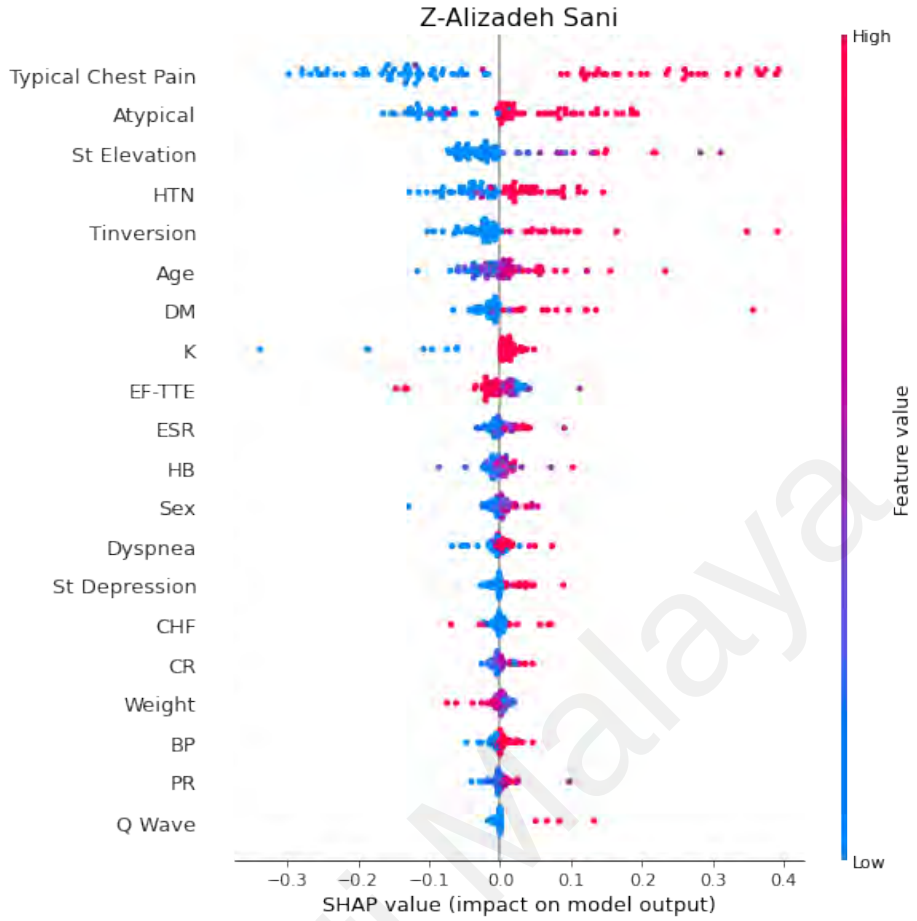
On the other hand, HD clinical records data sets explained the mortality rate and the prognosis of the patient having CVD. According to this data set, with an increase of one of the factors, the outcome will be increased mortality or worst prognosis except for the TTE-EF. Any increase means a better prognosis, and the sex where if the sex is female, the prognosis is better since females follow up more frequently (Adams Jr et al., 1999; Frazier et al., 2007; Ghali et al., 2002; O'Meara et al., 2007; Simon et al., 2001) (since it is red means it's of highest survival rate and the males are in blue, so they have more mortality rate). Also, we should consider the follow-up period since as it decreases, the mortality of CHF increases and the prognosis worsens (as the indicator change from blue to red, the follow-up period increases and the mortality rate decreases), as represented in Fig. 4.3 (d).



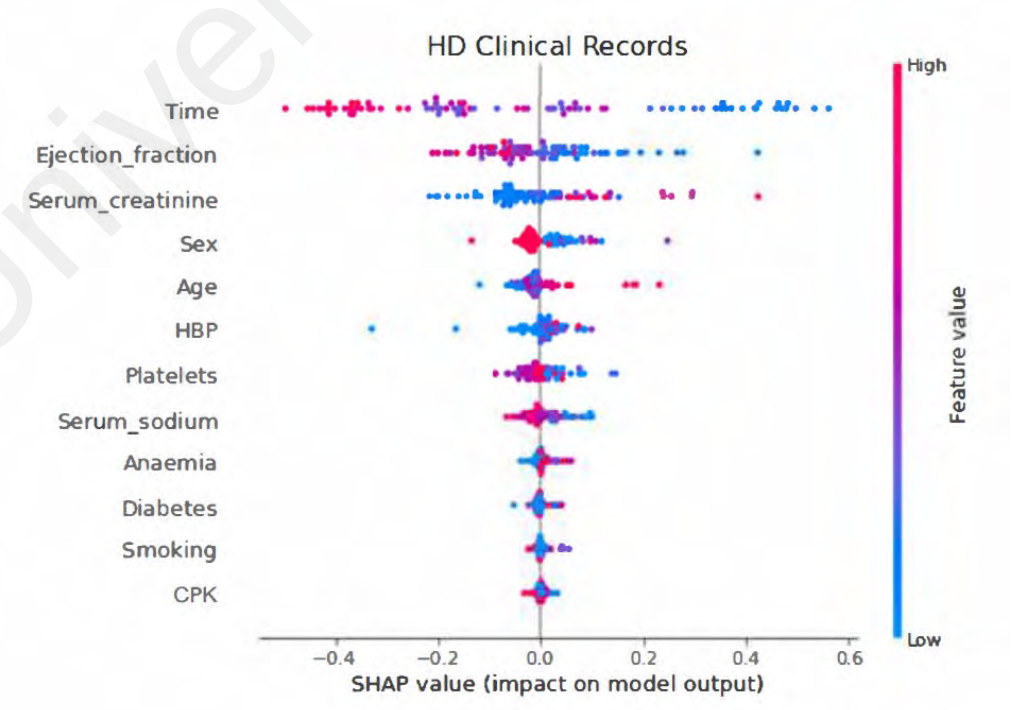
(a)



(b)



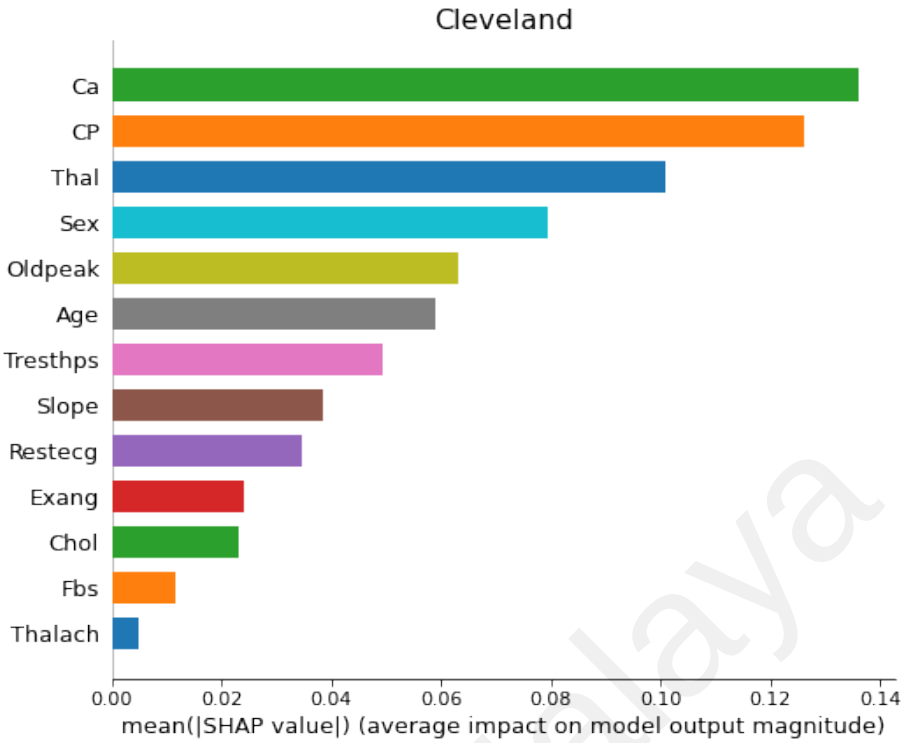
(c)



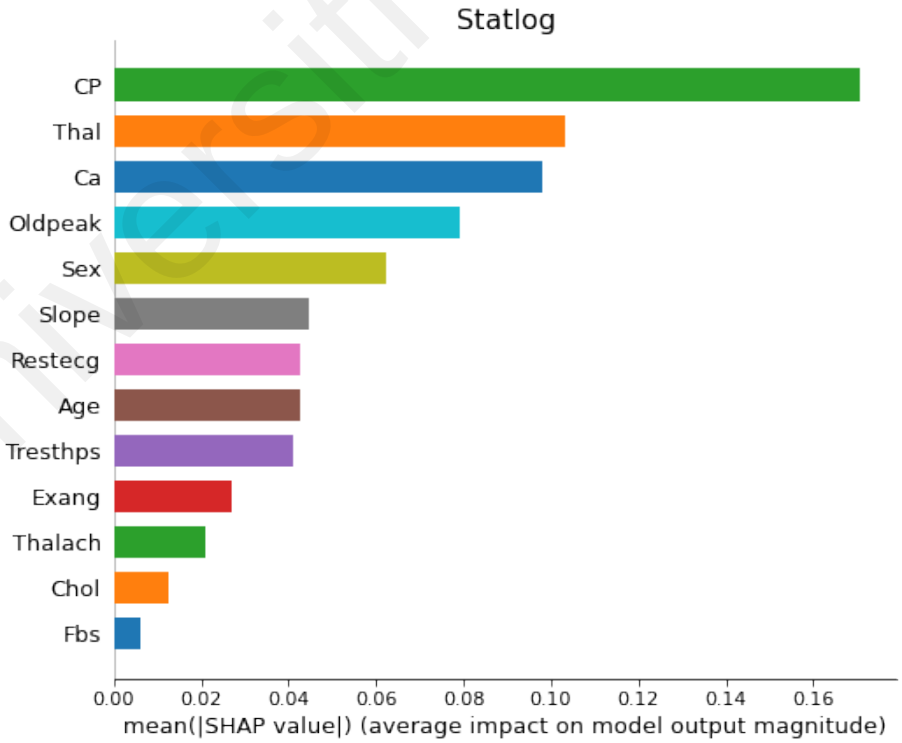
(d)

Figure 4.2: features rank based on SHAP values for SPFHD prediction of presence and patient survival of CVD. Red indicates a high value, blue is a low value for attributes, whereas SHAP values (negative or positive) reflect the directionality of the attributes. Negative SHAP values represent negative predictions (absence (a, b, c) and alive (d)). Conversely, positive SHAP values signify positive predictions (presence (a, b, c) and death event (d)).

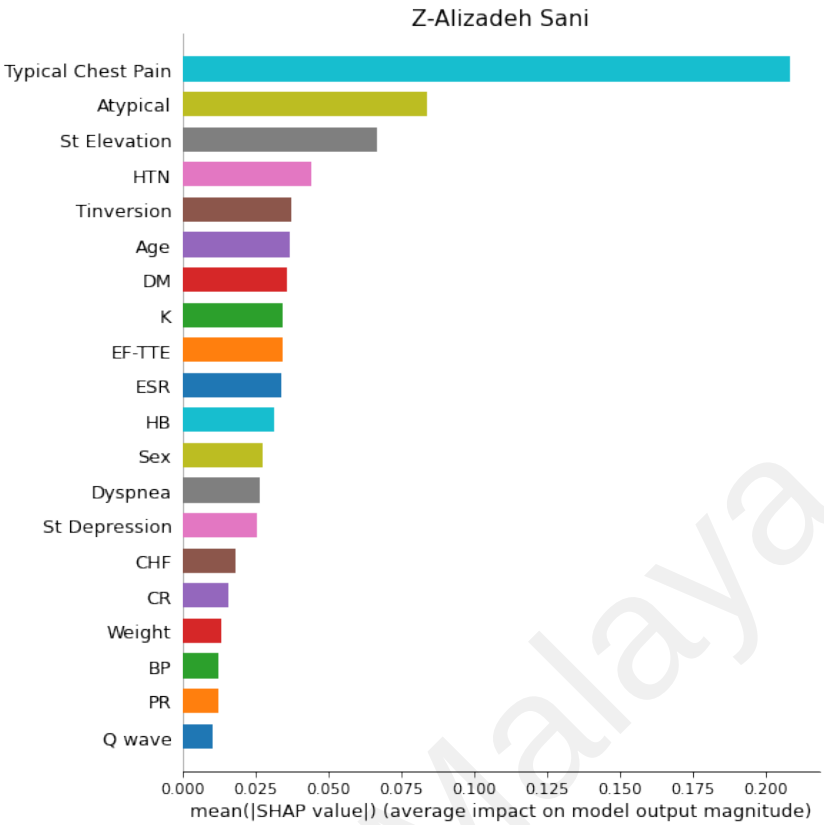
Moreover, the second illustration of the SHAP analysis (Fig. 4.4) provides general information about the features' importance on different datasets. It can be seen in Figure 4.4 the impact of each feature on different datasets. For instance, Ca, CP, and Thal have more impact on the model output, while Thalach, Fbs, Chol, and Exang have small contributions. Likewise, the Statlog dataset shares the same best and least features that impact the model's output with different means, as shown in Fig. 4.4 (a,b). In addition, Typical and Atypical chest pain have the highest impact on the model's output in the Z-alizadeh Sani dataset presented in Fig. 4.4 (c) (same feature 'CP' in Cleveland and statlog). The feature importance for the HD clinical dataset is illustrated in Fig. 4.4 (d), where time, ejection fraction, and serum creatinine contribute the most to the model's output, while anemia, diabetes, smoking, and CPK have a minimal impact on the model's output.



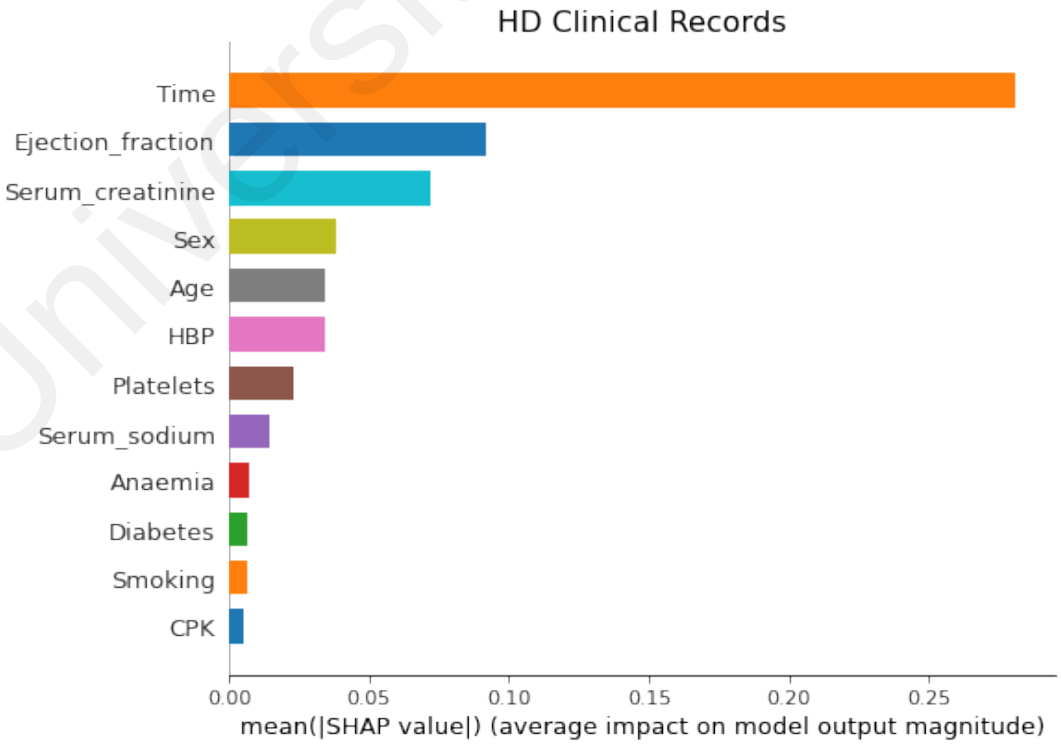
(a)



(b)



(c)



(d)

Figure 4.3: The effect of each attribute on SPFHD's output according to the SHAP framework where each figure refers to the ranking of the attribute effectiveness on different datasets (a) Cleveland, (b) Statlog, (c) Z-AliZadeh Sani, and (d) HD Clinical Records.

Finally, in the future, to increase the accuracy of the SPFHD model to assist in the diagnosis and calculating the prognosis of CVD, it may include add-on components like pro-BNP and C-troponin T (Anand et al., 2003; Berger et al., 2002; Koglin et al., 2001; Stanek et al., 2001) and other features like the number of hospitalization. The diagnosis of CVD happens when the patient is inpatient or outpatient since these terms affect the mortality rate (Taylor et al., 2019). The outcomes showed that models lacking the eight best characteristics had a lower predictive performance as measured by the average Acc (99.01 vs. 98.02) and MCC (0.98 vs. 0.96), among other performance metrics for HD clinical records. The features are selected after applying SHAP analysis to investigate the most contributing factors to SPFHD's output. The features chosen for all the datasets are shown in Table 4.11. These findings showed how crucial these features (selected by SHAP) are to SPFHD's ability to make accurate predictions. The results after applying the most key features highlighted by SHAP on the four datasets are shown in Table 4.12.

Table 4.11: The selected features after applying the SHAP analysis for the four datasets.

Dataset	Selected Features
Cleveland and Statlog	Ca, CP, Thal, Sex, Age, Oldpeak, Tresthps, Restecg, Slope
Z-Alizadeh Sani	Typical chest pain, Atypical, ST elevation, HTN, T inversion, Age, DM, K, EF-TTE, ESR, HB, Sex, Dyspnea, St depression
HD clinical Records	Time, Ejection fraction, Serum creatinine, Sex, Age, HBP, Platelets, Serum sodium

Table 4.12: The comparison between the proposed model with optimal features selected by SHAP.

Model	Cleveland					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	98.93	98.10	99.33	98.69	98.67	0.98
FS-SPFHD	99.47	99.35	99.33	99.33	99.56	0.99
Model	**Statlog**					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	99.02	99.24	98.25	98.72	99.48	0.98
FS-SPFHD	99.21	99.35	98.64	98.97	99.52	0.99
Model	**Z-Alizadeh Sani**					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	98.85	98.69	99.09	98.87	98.60	0.98
FS-SPFHD	100	100	100	100	100	1
Model	**HD clinical Records**					
	ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
SPFHD	98.02	98.00	97.95	97.96	98.14	0.96
FS-SPFHD	99.01	98.50	99.49	98.98	98.66	0.98

4.4 Statistical Analysis

Two-step statistical tests have been conducted for statistical verification between the proposed model and the other ML models to show the significant difference between classifiers across multiple datasets. First, an omnibus test utilizing the Friedman rank is performed as recommended in (Demšar, 2006). If differences in classifier performance can be identified, then a Friedman post hoc test is conducted. The Friedman test analyses

the hierarchy of benchmarked classifiers, whereas the Iman-Davenport test determines whether at least one classifier has a considerable advantage over others. Finally, a pair-wise test utilizing the Friedman post hoc with the corresponding p-value is performed for multiple comparisons after identifying such a difference. In the Friedman test, the classifier optimized by BO is considered since BO-SPFHD achieved the highest improvement rate among the three HPO methods. Regarding the Friedman post hoc test, a comparison to the reference (BO-SPFHD) is considered. The BO-SPFHD is chosen as a control classifier against comparing other base learners, including RFC, ETC, GBC, XGB, and LGBM optimized using BO. The significance of a difference is determined by a p-value that must be less than the threshold (0.05). Table 4.13 displays the mean ACC value, Friedman's average rank, and the Iman-Davenport test's p-value. It should be noted that the better the classifier, the lower its rank. The evidence is shown in Table 4.8, demonstrating that the SPFHD algorithm is the best method because it has the lowest rank (Friedman Rank=1). The p-value = 0.033, indicating a substantial difference (p-value < 0.05) between at least two benchmarked methods, which indicates that the null hypothesis assuming comparable performance across all classifiers can be rejected. In addition, once the null hypothesis is rejected, the Friedman post hoc test is used to assess the performance differences between the pairs. The results of the statistical comparison between the pairings are shown in Table 4.14. Notably, the differences in performance between the proposed algorithm and all basic learners are highly significant (p-value < 0.05). These findings confirmed that SPFHD's stacking method could learn a high-level classifier more efficiently than five base learners.

Table 4.13: The results of the Friedman rank and Iman-Davenport tests, as well as the mean value of the accuracy (%).

Model	Cleveland	Statlog	Z- Alizadeh Sani	HD Clinical Records	Friedman Rank	Iman-Davenport p-value
BO-RFC	97.60	98.68	94.71	96.32	5	0.033226
BO-ETC	97.87	98.98	97.24	96.81	3.13	
BO-GBC	97.33	97.66	95.17	95.12	5.75	
BO-XGB	98.14	98.15	95.86	97.53	3.5	
BO-LGBM	97.87	98.94	98.16	97.79	2.63	
BO-SPFHD	**98.93**	**99.02**	**98.85**	**98.49**	**1**	

Table 4.14: Comparison of the proposed method and other classifiers with respect to Friedman's post hoc test

Comparison	p-value (post hoc)
Proposed VS BO-RFC	0.030
Proposed VS BO-ETC	0.032
Proposed VS BO-GBC	0.006
Proposed VS BO-XGB	0.022
Proposed VS BO-LGBM	0.035

4.5 Comparative Analysis

A performance evaluation and comparison of the developed method to current methods. Using the same dataset to evaluate predicted performance is typically a more objective and bias-free method. Comparing SPFHD to state-of-the-art predictors across the four datasets, namely Cleveland, Statlog, Z-Alizadeh Sani, and HD clinical records,

presented in Tables 4.13, 4.14, 4.15, and 4.16, respectively, reveal that SPFHD has superior predictive performance in terms of ACC, PPV, TPR, F1, SPC, and MCC for HD prediction when compared to the other approaches including ML and DL models. For example, the developed stack-ensemble model achieved better results than SMOTE+CNN (Umer et al., 2022) (refer to table 4.14) and SMOTE + Deep Learning (Waqar et al., 2021) (refer to table 4.11); the ACC values achieved using SPFHD are 99.01% and 99.47% as compared to SMOTE+CNN (92.63%) and SMOTE+ deep learning (96%) respectively, where DL entails more data and memory to train. In addition, the proposed model achieved higher results than DBMRITLBO-ANN (95.41%) and ANN + RF (95.08) (refer to table 4.13), as the model focused on ANN, which is more likely to overfit. Generally, the models developed using single models did not perform well, as shown in Tables 4.14 and 4.15. Finally, the ACC attained by SPFHD was around 2% higher than other models across all four datasets, indicating that SPFHD is superior to earlier methods. Notably, only SPFHD on HD clinical records dataset achieved consistent results compared to the models, achieving better ACC, PPV, TPR, and F1 by 1.81%, 4.1%, 5.19%, and 4.68%, respectively, than the second model. Therefore, these findings suggested that SHFPD is a reliable and stable predictor of HD diagnosis.

Table 4.15: Performance comparison between the proposed method and previous work on the Cleveland dataset.

Author	Method	Performance Evaluation					
		ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
(Tama et al., 2020)	Two-tier ensemble PSO based FS	85.71	-	-	-	86.49	-
(Fitriyani et al., 2020)	DBSCAN + SMOTE-ENN + XGBoost	98.4	98.57	98.33	98.33	98.32	0.97
(Nilashi et al., 2020)	KNN+SOM+PCA+ Fuzzy SVM	96.86	-	96.66	94.35	-	-
(Thanga Selvi & Muthulakshmi, 2021)	DBMRITLBO-ANN	95.41	-	97.33	93.23	95.74	-
(Waqar et al., 2021)	SMOTE + Deep Learning	96	96.1	95.7	-	95.7	-
(Vivekanandan & Narayanan, 2019)	DE-Cox regression	91	-	-	-	-	-
(Kibria & Matin, 2022)	ANN + RF	95.08	95	95	-	95	-
(Shan et al., 2022)	MGOHBO-KELM	81.85	-	82..76	85.47	-	0.6513
(Asadi et al., 2021)	MOPSO-RF	85.21	-	-	-	-	-
Proposed Method 2	**SPFHD**	**99.47**	99.35	**99.33**	**99.33**	**99.56**	**0.99**

Table 4.16: Performance comparison between the proposed method and previous work on the Statlog dataset.

Author	Method	Performance Evaluation					
		ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
(Tama et al., 2020)	Two-tier ensemble PSO based FS	93.55	-	-	-	91.67	-
(Fitriyani et al., 2020)	DBSCAN + SMOTE-ENN + XGBoost	95.9	97.1	94.6	95.4	95.3	0.92
(Nilashi et al., 2020)	KNN+SOM+PCA+ Fuzzy SVM	97.87	-	96.97	96.97	-	-
(Shan et al., 2022)	MGOHBO-KELM	75.91	-	58.02	86.99	-	0.45
(Asadi et al., 2021)	MOPSO-RF	88.26	-	-	-	-	
Proposed Method 1	**Inf-FSs+BO+IWRF**	**0.983**	**0.986**	**0.972**	**0.979**	**0.991**	**0.966**
Proposed Method 2	**SPFHD**	**99.21**	**99.35**	**98.64**	**98.97**	**99.52**	**0.99**

Table 4.17: Performance comparison between the proposed method and previous work on the Z-Alizadeh Sani dataset.

Author	Method	Performance Evaluation					
		ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
(Tama et al., 2020)	Two-tier ensemble PSO based FS	98.13	-	-	-	96.6	-
(Abdar et al., 2019)	N2Genetic-nuSVM	93.08	-	-	-	91.51	-
(Yuvalı et al., 2022)	RS-LR	92.4	89.5	91.9	-	90.7	-
(A. Gupta et al., 2022)	FAMD + BBA + RF-ET	97.37	-	98.15	95.45	-	0.45
Proposed Method 2	**SPFHD**	**100**	**100**	**100**	**100**	**100**	**1**

Table 4.18: Performance comparison between the proposed method and previous work on the HD clinical records dataset.

Author	Method	Performance Evaluation					
		ACC	**PPV**	**TPR**	**F1**	**SPC**	**MCC**
(Ishaq et al., 2021)	SMOTE + RF + ET	92.6	93	93	93	-	-
(Almazroi, 2022)	DT	80	78.94	65.21	-	71.4	-
(Umer et al., 2022)	SMOTE+CNN	92.63	92.81	93.99	-	93.4	-
Proposed Method 1	**Inf-FSs+BO+IWRF**	**97.2**	**94.4**	**94.3**	**94.3**	**98.2**	**0.922**
Proposed Method 2	**SPFHD**	**99.01**	**98.50**	**99.49**	**98.98**	**98.66**	**0.98**

CHAPTER 5: CONCLUSION AND FUTURE WORK

This chapter fills the gaps and supplements existing research in the literature concerning the evolving issue of CVD detection. Hence, two methodologies were proposed, the first on the algorithm level by selecting a higher weight for the minor class, while the other is on the data level to balance the two classes by generating new data samples from the minor class. Moreover, recommendations for further enhancements will be proposed as part of future work in the current study.

5.1 Conclusion

The CVD detection model serves as a valuable aid for cardiologists by categorizing each sample into its respective class, distinguishing between CVD and non-CVD. The ML emerges as a contemporary and efficacious technique for disease diagnosis, with various architectures and methodologies proposed to classify individuals into different classes precisely. The binary classification task of CVD detection involves categorizing individuals as having CVD or non-CVD. Class imbalance in the number of CVD and non-CVD samples introduces challenges in reducing detection accuracy. Additionally, the presence of outliers and missing data further complicates accurate predictions, as samples exhibit dual properties of both CVD and non-CVD classes.

Furthermore, this thesis extends its focus to multiclassification for severity classification, aiming to categorize individuals into different severity levels of CVD. A critical analysis of the impact of class imbalance data on the training process is conducted, exploring the potential of ML structures and other methods at both the algorithmic and data levels to address the inherent differences between CVD and non-CVD samples. A balancing factor is derived from the ratio of CVD to non-CVD samples, contributing to a more balanced training process.

Moreover, CVD detection traditionally involves heart imaging or ECG, but this thesis specifically concentrates on the detection of CVD through clinical data. The hybrid ML model developed in the study undergoes further optimization of hyperparameters to enhance the prediction accuracy of CVD. The optimization is carried out using Tree-Structured Parzen Estimator Bayesian Optimization (TPE-BO), which is compared against other optimization methods such as PSO, GA, and RS. Various datasets, including Cleveland, statlog, Z-Alizadeh, and heart disease clinical records, are employed to validate and evaluate the performance of the proposed methods. Implementation and training of several CVD detection models are conducted using Python frameworks like scikit-learn and TensorFlow, with Python utilized for dataset preprocessing, including normalization and scaling. Finally, the evaluation of ML model performance is based on classification accuracy and misclassification error rate, showcasing the effectiveness of the proposed methods across diverse and complex real-world datasets.

In pursuit of the first objective, an exhaustive inquiry into machine learning-driven methodologies for the detection of cardiovascular diseases utilizing clinical data classification was conducted. This investigation aimed to establish a ranking system for identifying optimal models applicable to cardiovascular disease detection. Subsequently, an IWRF approach was devised to address the challenge of data imbalance at the algorithmic level. The contribution of this objective lies in developing and improving the Random Forest algorithm to handle data imbalance more effectively at the algorithm level, which can be applied to various other applications facing similar data challenges. The proposed model improved the performance of CVD detection by 3.62% and 4.82% for the Statlog dataset and 6.3% and 11.98% for HD clinical records in terms of accuracy and f1-score, respectively, as compared to the recently published works by reaching values of 98.3% and 99.1% for accuracy and f1-score for Statlog dataset. As well as reaching values of 97.2% and 98.2% accuracy and f-score for HD clinical records dataset.

For the second objective, a method based on conditional variational auto-encoder was formulated to tackle the data imbalance concern at the data level. This involved generating new samples from the minor class that adhere to the original data distribution, thereby achieving dataset equilibrium. The introduced model demonstrated notable efficacy in addressing the imbalanced data issue at the data level and underwent validation using imbalanced datasets for CVD detection. The results showed that the proposed SPFHD model outperformed the state-of-art methods over four datasets, achieving higher f1-score of 4.68 %, 4.55 %, 2 %, and 1 % for HD clinical, Z-Alizadeh Sani, Statlog, and Cleveland, respectively. The contribution of this objective lies in the development of the CVAE model, which had not been applied to these types of applications before, and it effectively solves the data imbalance issue at the data level, offering a novel approach that can be extended to other fields facing similar challenges The enhanced performance of SPFHD can be attributed to the new balancing model (CVAE) and hyperparameter optimization.

For the third objective, a cardiovascular disease detection system was established employing a hybrid machine learning model designed to enhance prediction accuracy. This involved the development of a model architecture (IWRF and the CVAE-SPFHD) and subsequent hyperparameter optimization. The proposed model was then subjected to a comprehensive statistical analysis, comparing its performance with existing models for validation and evaluation. The significance and contribution of this objective lie in developing a multi-level approach that addresses critical challenges from data preprocessing to feature selection and optimization. The system achieves greater accuracy and reliability by integrating algorithm and data-level methodologies such as data balancing, feature ranking, and hyperparameter optimization. This holistic approach enhances prediction performance for cardiovascular disease detection and offers a framework that can be applied to various domains requiring robust ML solutions.

In the final objective, the study delves into the proposed learning mechanisms, emphasizing the most influential features that empower the model to yield accurate CVD prediction outcomes. A thorough investigation leveraging the SHAP framework was conducted to gain a deeper insight into the model's inner workings and interpretability. This model interpretation exercise elucidated the most pivotal features and parameters for each dataset. By identifying these key features, the proposed model (SPFHD) demonstrated a heightened capacity to detect heart diseases, thereby offering a more effective approach based on these discerning features rather than relying on alternative ones.

5.2 Future work

Despite the commendable performance exhibited by the proposed methodologies, IWRF and SPFHD, in predicting the presence of CVD and the survival of patients, certain limitations warrant consideration. The primary constraint lies in the relatively constrained nature of the training dataset, particularly concerning features. The inclusion of additional features is deemed essential for enhancing diagnostic accuracy, necessitating a broader spectrum of relevant patient characteristics. The study acknowledges that the predictive performance for patient status falls short of complete satisfaction due to these constraints.

Future endeavors will center around the expansion of this work, primarily concentrating on the collection of new data encompassing a more comprehensive array of features and a larger patient cohort. Specifically, the incorporation of vital features like pro-BNP and C-troponin T, anticipated to become available in subsequent datasets, will be pivotal in improving diagnostic accuracy. Additionally, outlier detection and removal techniques will be implemented to ensure the quality of the data used for training. By identifying and removing data points that deviate significantly from the norm, the model’s robustness and predictive reliability will be further enhanced. Beyond data collection and

feature expansion, future work will explore advanced techniques for detecting and managing data outliers, which can significantly impact model performance if left unaddressed. Outlier detection will be critical in ensuring that the model's predictions are not skewed by anomalous data points, and methods such as Isolation Forest and robust Z-scores will be considered to handle this issue effectively.

Moreover, this work's scope is poised for enlargement, not only through the incorporation of new features but also by encompassing a more extensive segment of patients. The stable and robust SPFHD framework and IWRF methodology hold promise for facile adaptation and extension to diverse survival and severity identification tasks, potentially extending their applicability to tasks such as heart disease severity level identification and diabetes prediction. Further research may also investigate the use of deep learning models or transfer learning to improve predictive performance across different heart disease subtypes and patient populations. Additionally, there is potential for integrating multimodal data sources, such as imaging and genetic data, to provide more holistic predictions in future iterations of the model. This will significantly enhance the model's adaptability and accuracy in broader clinical applications.

REFERENCES

Abdar, M., Książek, W., Acharya, U. R., Tan, R.-S., Makarenkov, V., & Pławiak, P. (2019). A new machine learning technique for an accurate diagnosis of coronary artery disease. *Computer methods and programs in biomedicine*, *179*, 104992.

Abdellatif, A., Abdellatef, H., Kanesan, J., Chow, C.-O., Chuah, J. H., & Gheni, H. M. (2022). An effective heart disease detection and severity level classification model using machine learning and hyperparameter optimization methods. *IEEE access*, *10*, 79974-79985.

Abdellatif, A., Abdellatef, H., Kanesan, J., Onn, C. C., Chuah, J. H., & Gheni, H. M. (2022). Improving the Heart Disease Detection and Patients' Survival using Supervised Infinite Feature Selection and Improved Weighted Random Forest. *IEEE Access*.

Abdellatif, A., Mubarak, H., Abdellatef, H., Kanesan, J., Abdelltif, Y., Chow, C.-O., Chuah, J. H., Gheni, H. M., & Kendall, G. (2024). Computational detection and interpretation of heart disease based on conditional variational auto-encoder and stacked ensemble-learning framework. *Biomedical Signal Processing and Control*, *88*, 105644.

Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, *2*(4), 433-459.

Acharya, U. R., Fujita, H., Oh, S. L., Hagiwara, Y., Tan, J. H., Adam, M., & Tan, R. S. (2019). Deep convolutional neural network for the automated diagnosis of congestive heart failure using ECG signals. *Applied Intelligence*, *49*(1), 16-27.

Adams Jr, K. F., Sueta, C. A., Gheorghiade, M., O'Connor, C. M., Schwartz, T. A., Koch, G. G., Uretsky, B., Swedberg, K., McKenna, W., & Soler-Soler, J. (1999). Gender differences in survival in advanced heart failure: insights from the FIRST study. *Circulation*, *99*(14), 1816-1821.

Aggarwal, C. C., & Reddy, C. K. (2014). Data clustering. *Algorithms and applications. Chapman&Hall/CRC Data mining and Knowledge Discovery series, Londra*.

Ahmad, G. N., Fatima, H., & Saidi, A. S. (2022). Efficient Medical Diagnosis of Human Heart Diseases using Machine Learning Techniques with and without GridSearchCV. *IEEE Access*.

Ahmad, T., Munir, A., Bhatti, S. H., Aftab, M., & Raza, M. A. (2017). Survival analysis of heart failure patients: A case study. *PloS one*, *12*(7), e0181001.

Ahsan, M. M., & Siddique, Z. (2022). Machine learning-based heart disease diagnosis: A systematic literature review. *Artificial Intelligence in Medicine*, *128*, 102289.

Aïmeur, E., Brassard, G., & Gambs, S. (2013). Quantum speed-up for unsupervised learning. *Machine learning*, *90*(2), 261-287.

Al-Delaimy, W. K., Merchant, A. T., Rimm, E. B., Willett, W. C., Stampfer, M. J., & Hu, F. B. (2004). Effect of type 2 diabetes and its duration on the risk of peripheral arterial disease among men. *The American journal of medicine*, *116*(4), 236-240.

Albert, A. J., Murugan, R., & Sripriya, T. (2023). Diagnosis of heart disease using oversampling methods and decision tree classifier in cardiology. *Research on Biomedical Engineering*, *39*(1), 99-113.

Ali, F., El-Sappagh, S., Islam, S. R., Kwak, D., Ali, A., Imran, M., & Kwak, K.-S. (2020). A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion. *Information Fusion*, *63*, 208-222.

Ali, L., Niamat, A., Khan, J. A., Golilarz, N. A., Xingzhong, X., Noor, A., Nour, R., & Bukhari, S. A. C. (2019). An optimized stacked support vector machines based expert system for the effective prediction of heart failure. *IEEE Access*, *7*, 54007-54014.

Ali, M. M., Paul, B. K., Ahmed, K., Bui, F. M., Quinn, J. M., & Moni, M. A. (2021). Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison. *Computers in Biology and Medicine*, *136*, 104672.

Alipanahi, B., Delong, A., Weirauch, M. T., & Frey, B. J. (2015). Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature biotechnology*, *33*(8), 831-838.

Aljaaf, A. J., Al-Jumeily, D., Hussain, A. J., Dawson, T., Fergus, P., & Al-Jumaily, M. (2015). Predicting the likelihood of heart failure with a multi level risk assessment using decision tree. 2015 Third International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE),

Almazroi, A. A. (2022). Survival prediction among heart patients using machine learning techniques. *Mathematical Biosciences and Engineering*, *19*(1), 134-145.

Almdal, T., Scharling, H., Jensen, J. S., & Vestergaard, H. (2004). The independent effect of type 2 diabetes mellitus on ischemic heart disease, stroke, and death: a population-based study of 13 000 men and women with 20 years of follow-up. *Archives of internal medicine*, *164*(13), 1422-1426.

Anand, I. S., Fisher, L. D., Chiang, Y.-T., Latini, R., Masson, S., Maggioni, A. P., Glazer, R. D., Tognoni, G., & Cohn, J. N. (2003). Changes in brain natriuretic peptide and norepinephrine over time and mortality and morbidity in the Valsartan Heart Failure Trial (Val-HeFT). *Circulation*, *107*(9), 1278-1283.

Arabasadi, Z., Alizadehsani, R., Roshanzamir, M., Moosaei, H., & Yarifard, A. A. (2017). Computer aided decision making for heart disease detection using hybrid neural network-Genetic algorithm. *Computer methods and programs in biomedicine*, *141*, 19-26.

Arasi, M. A., El-Horbaty, E.-S. M., & El-Sayed, A. (2018). Classification of dermoscopy images using naive bayesian and decision tree techniques. 2018 1st Annual International Conference on Information and Sciences (AiCIS),

Asadi, S., Roshan, S., & Kattan, M. W. (2021). Random forest swarm optimization-based for heart diseases diagnosis. *Journal of Biomedical Informatics*, *115*, 103690.

Atkov, O. Y., Gorokhova, S. G., Sboev, A. G., Generozov, E. V., Muraseyeva, E. V., Moroshkina, S. Y., & Cherniy, N. N. (2012). Coronary heart disease diagnosis by artificial neural networks including genetic polymorphisms and clinical parameters. *Journal of cardiology*, *59*(2), 190-194.

Bai, W., Sinclair, M., Tarroni, G., Oktay, O., Rajchl, M., Vaillant, G., Lee, A. M., Aung, N., Lukaschuk, E., & Sanghvi, M. M. (2018). Automated cardiovascular magnetic resonance image analysis with fully convolutional networks. *Journal of Cardiovascular Magnetic Resonance*, *20*(1), 1-12.

Bashar, S. K., Han, D., Zieneddin, F., Ding, E., Fitzgibbons, T. P., Walkey, A. J., McManus, D. D., Javidi, B., & Chon, K. H. (2020). Novel density poincare plot based machine learning method to detect atrial fibrillation from premature atrial/ventricular contractions. *IEEE Transactions on Biomedical Engineering*, *68*(2), 448-460.

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, *6*(1), 20-29.

Bejjanki, K. K., Gyani, J., & Gugulothu, N. (2020). Class imbalance reduction (CIR): a novel approach to software defect prediction in the presence of class imbalance. *Symmetry*, *12*(3), 407.

Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. *Large-scale kernel machines*, *34*(5), 1-41.

Benjamin, E. J., Muntner, P., Alonso, A., Bittencourt, M. S., Callaway, C. W., Carson, A. P., Chamberlain, A. M., Chang, A. R., Cheng, S., & Das, S. R. (2019). Heart disease and stroke statistics—2019 update: a report from the American Heart Association. *Circulation*, *139*(10), e56-e528.

Berger, R., Huelsman, M., Strecker, K., Bojic, A., Moser, P., Stanek, B., & Pacher, R. (2002). B-type natriuretic peptide predicts sudden death in patients with chronic heart failure. *Circulation*, *105*(20), 2392-2397.

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. 25th annual conference on neural information processing systems (NIPS 2011),

Beunza, J.-J., Puertas, E., García-Ovejero, E., Villalba, G., Condes, E., Koleva, G., Hurtado, C., & Landecho, M. F. (2019). Comparison of machine learning algorithms for clinical event prediction (risk of coronary heart disease). *Journal of biomedical informatics*, *97*, 103257.

Bhatt, C. M., Patel, P., Ghetia, T., & Mazzeo, P. L. (2023). Effective heart disease prediction using machine learning techniques. *Algorithms*, *16*(2), 88.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

Bizopoulos, P., & Koutsouris, D. (2018). Deep learning in cardiology. *IEEE reviews in biomedical engineering*, *12*, 168-193.

Blagus, R., & Lusa, L. (2013). SMOTE for high-dimensional class-imbalanced data. *BMC bioinformatics*, *14*(1), 1-16.

Bonnefont-Rousselot, D. (2016). Resveratrol and cardiovascular diseases. *Nutrients*, *8*(5), 250.

Bonyadi, M. R., & Michalewicz, Z. (2017). Particle swarm optimization for single objective continuous space problems: a review. *Evolutionary computation*, *25*(1), 1-54.

Booker, L. B., Goldberg, D. E., & Holland, J. H. (1989). Classifier systems and genetic algorithms. *Artificial intelligence*, *40*(1-3), 235-282.

Bourlard, H., & Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, *59*(4), 291-294.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Budholiya, K., Shrivastava, S. K., & Sharma, V. (2020). An optimized XGBoost based diagnostic system for effective prediction of heart disease. *Journal of King Saud University-Computer and Information Sciences*.

Charoenkwan, P., Schaduangrat, N., Moni, M. A., Manavalan, B., & Shoombuatong, W. (2022). SAPPHIRE: A stacking-based ensemble learning framework for accurate prediction of thermophilic proteins. *Computers in Biology and Medicine*, 105704.

Chaudhuri, A. K., Das, S., & Ray, A. (2024). An Improved Random Forest Model for Detecting Heart Disease. In *Data-Centric AI Solutions and Emerging Technologies in the Healthcare Ecosystem* (pp. 143-164). CRC Press.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357.

Chen, S., Webb, G. I., Liu, L., & Ma, X. (2020). A novel selective naïve Bayes algorithm. *Knowledge-Based Systems*, *192*, 105361.

Chen, T.-E., Yang, S.-I., Ho, L.-T., Tsai, K.-H., Chen, Y.-H., Chang, Y.-F., Lai, Y.-H., Wang, S.-S., Tsao, Y., & Wu, C.-C. (2016). S1 and S2 heart sound recognition using deep neural networks. *IEEE Transactions on Biomedical Engineering*, *64*(2), 372-380.

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., & Chen, K. (2015). Xgboost: extreme gradient boosting. *R package version 0.4-2*, *1*(4), 1-4.

Cheng, K., Zhang, C., Yu, H., Yang, X., Zou, H., & Gao, S. (2019). Grouped SMOTE with noise filtering mechanism for classifying imbalanced data. *IEEE Access*, *7*, 170668-170681.

Cheriyan, J., O'Shaughnessy, K. M., & Brown, M. J. (2010). Primary prevention of CVD: treating hypertension. *BMJ Clin Evid*, *2010*, 0214.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Choi, E., Schuetz, A., Stewart, W. F., & Sun, J. (2017). Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, *24*(2), 361-370.

Cioffi, R., Travaglioni, M., Piscitelli, G., Petrillo, A., & De Felice, F. (2020). Artificial intelligence and machine learning applications in smart production: Progress, trends, and directions. *Sustainability*, *12*(2), 492.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of machine learning research*, *12*(ARTICLE), 2493− 2537.

Dalen, J. E., Alpert, J. S., Goldberg, R. J., & Weinstein, R. S. (2014). The epidemic of the 20th century: coronary heart disease. *The American journal of medicine*, *127*(9), 807-812.

DeCastro-García, N., Muñoz Castañeda, Á. L., Escudero García, D., & Carriegos, M. V. (2019). Effect of the Sampling of a Dataset in the Hyperparameter Optimization Phase over the Efficiency of a Machine Learning Algorithm. *Complexity*, *2019*.

Deekshatulu, B., & Chandra, P. (2013). Classification of heart disease using k-nearest neighbor and genetic algorithm. *Procedia technology*, *10*, 85-94.

Dekamin, A., & Sheibatolhamdi, A. (2017). A data mining approach for coronary artery disease prediction in Iran. *Journal of Advanced Medical Sciences and Applied Technologies*, *3*(1), 29-38.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, *7*, 1-30.

Dilsizian, M. E., & Siegel, E. L. (2018). Machine meets biology: a primer on artificial intelligence in cardiology and cardiac imaging. *Current cardiology reports*, *20*(12), 1-7.

Dubey, A., Gupta, U., & Jain, S. (2021). Medical data clustering and classification using TLBO and machine learning algorithms. *Computers, Materials and Continua*, *70*(3), 4523-4543.

Eckel, R. H., York, D. A., Rössner, S., Hubbard, V., Caterson, I., St. Jeor, S. T., Hayman, L. L., Mullis, R. M., & Blair, S. N. (2004). Prevention Conference VII: Obesity, a worldwide epidemic related to heart disease and stroke: executive summary. *Circulation*, *110*(18), 2968-2975.

Elshawi, R., Maher, M., & Sakr, S. (2019). Automated machine learning: State-of-the-art and open challenges. *arXiv preprint arXiv:1906.02287*.

Faieq, A. K., & Mijwil, M. M. (2022). Prediction of heart diseases utilising support vector machine and artificial neural network. *Indonesian Journal of Electrical Engineering and Computer Science*, *26*(1), 374-380.

Fitriyani, N. L., Syafrudin, M., Alfian, G., & Rhee, J. (2020). HDPM: an effective heart disease prediction model for a clinical decision support system. *IEEE Access*, *8*, 133034-133050.

Franklin, S. S., Larson, M. G., Khan, S. A., Wong, N. D., Leip, E. P., Kannel, W. B., & Levy, D. (2001). Does the relation of blood pressure to coronary heart disease risk change with aging? The Framingham Heart Study. *Circulation*, *103*(9), 1245-1249.

Frazier, C. G., Alexander, K. P., Newby, L. K., Anderson, S., Iverson, E., Packer, M., Cohn, J., Goldstein, S., & Douglas, P. S. (2007). Associations of gender and etiology with outcomes in heart failure with systolic dysfunction: a pooled analysis of 5 randomized control trials. *Journal of the American College of Cardiology*, *49*(13), 1450-1458.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

Garcia-Carretero, R., Vigil-Medina, L., Mora-Jimenez, I., Soguero-Ruiz, C., Barquero-Perez, O., & Ramos-Lopez, J. (2020). Use of a K-nearest neighbors model to predict the development of type 2 diabetes within 2 years in an obese, hypertensive population. *Medical & biological engineering & computing*, *58*(5), 991-1002.

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine learning*, *63*(1), 3-42.

Ghali, J. K., Piña, I. L., Gottlieb, S. S., Deedwania, P. C., & Wikstrand, J. C. (2002). Metoprolol CR/XL in female patients with heart failure: analysis of the experience in Metoprolol Extended-Release Randomized Intervention Trial in Heart Failure (MERIT-HF). *Circulation*, *105*(13), 1585-1591.

Goldberg, D. E., Korb, B., & Deb, K. (1989). Messy genetic algorithms: Motivation, analysis, and first results. *Complex systems*, *3*(5), 493-530.

Golpour, P., Ghayour-Mobarhan, M., Saki, A., Esmaily, H., Taghipour, A., Tajfard, M., Ghazizadeh, H., Moohebati, M., & Ferns, G. A. (2020). Comparison of support vector machine, naïve Bayes and logistic regression for assessing the necessity for coronary angiography. *International journal of environmental research and public health*, *17*(18), 6449.

Gotlibovych, I., Crawford, S., Goyal, D., Liu, J., Kerem, Y., Benaron, D., Yilmaz, D., Marcus, G., & Li, Y. (2018). End-to-end deep learning from raw sensor data: Atrial fibrillation detection using wearables. *arXiv preprint arXiv:1807.10707*.

Gu, Y., Yang, X., Tian, L., Yang, H., Lv, J., Yang, C., Wang, J., Xi, J., Kong, G., & Zhang, W. (2022). Structure-aware siamese graph neural networks for encounter-level patient similarity learning. *Journal of Biomedical Informatics*, *127*, 104027.

Guidi, G., Pettenati, M. C., Melillo, P., & Iadanza, E. (2014). A machine learning system to improve heart failure patient assistance. *IEEE journal of biomedical and health informatics*, *18*(6), 1750-1756.

Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., & Cuadros, J. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, *316*(22), 2402-2410.

Gupta, A., Kumar, R., Arora, H. S., & Raman, B. (2019). MIFH: A machine intelligence framework for heart disease diagnosis. *IEEE Access*, *8*, 14659-14674.

Gupta, A., Kumar, R., Arora, H. S., & Raman, B. (2022). C-CADZ: computational intelligence system for coronary artery disease detection using Z-Alizadeh Sani dataset. *Applied Intelligence*, *52*(3), 2436-2464.

Gupta, C., Saha, A., Reddy, N. S., & Acharya, U. D. (2022). Cardiac Disease Prediction using Supervised Machine Learning Techniques. Journal of Physics: Conference Series,

Gürbüz, E., & Kılıç, E. (2014). A new adaptive support vector machine for diagnosis of diseases. *Expert Systems*, *31*(5), 389-397.

Hannun, A. Y., Rajpurkar, P., Haghpanahi, M., Tison, G. H., Bourn, C., Turakhia, M. P., & Ng, A. Y. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature medicine*, *25*(1), 65-69.

Haq, A. U., Li, J. P., Memon, M. H., Nazir, S., & Sun, R. (2018). A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms. *Mobile Information Systems*, *2018*.

Harris, T., Cook, E. F., Kannel, W. B., & Goldman, L. (1988). Proportional hazards analysis of risk factors for coronary heart disease in individuals aged 65 or older: the Framingham Heart Study. *Journal of the American geriatrics society*, *36*(11), 1023-1028.

Hazan, E., Klivans, A., & Yuan, Y. (2017). Hyperparameter optimization: A spectral approach. *arXiv preprint arXiv:1706.00764*.

Helmstaedter, M., Briggman, K. L., Turaga, S. C., Jain, V., Seung, H. S., & Denk, W. (2013). Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature*, *500*(7461), 168-174.

Huang, Y., & Li, L. (2011). Naive Bayes classification algorithm based on small sample set. 2011 IEEE International conference on cloud computing and intelligence systems,

Hussain, L., Awan, I. A., Aziz, W., Saeed, S., Ali, A., Zeeshan, F., & Kwak, K. S. (2020). Detecting congestive heart failure by extracting multimodal features and employing machine learning techniques. *BioMed Research International*, *2020*.

Hussain, L., Lone, K. J., Awan, I. A., Abbasi, A. A., & Pirzada, J.-u.-R. (2022). Detecting congestive heart failure by extracting multimodal features with synthetic minority oversampling technique (SMOTE) for imbalanced data using robust machine learning techniques. *Waves in Random and Complex Media*, *32*(3), 1079-1102.

Hutter, F., Kotthoff, L., & Vanschoren, J. (2019). *Automated machine learning: methods, systems, challenges*. Springer Nature.

Ishaq, A., Sadiq, S., Umer, M., Ullah, S., Mirjalili, S., Rupapara, V., & Nappi, M. (2021). Improving the prediction of heart failure patients' survival using SMOTE and effective data mining techniques. *IEEE Access*, *9*, 39707-39716.

Jabbar, M., & Samreen, S. (2016). Heart disease prediction system based on hidden naïve bayes classifier. 2016 International Conference on Circuits, Controls, Communications and Computing (I4C),

James, P. A., Oparil, S., Carter, B. L., Cushman, W. C., Dennison-Himmelfarb, C., Handler, J., Lackland, D. T., LeFevre, M. L., MacKenzie, T. D., & Ogedegbe, O. (2014). 2014 evidence-based guideline for the management of high blood pressure in adults: report from the panel members appointed to the Eighth Joint National Committee (JNC 8). *Jama*, *311*(5), 507-520.

Janosi, A., Steinbrunn, W., Pfisterer, M., & Detrano, R. (1988). Heart disease data set. *The UCI KDD Archive*. https://archive.ics.uci.edu/ml/datasets/heart+disease

Kafai, M., & Eshghi, K. (2017). CROification: accurate kernel classification with the efficiency of sparse linear SVM. *IEEE transactions on pattern analysis and machine intelligence*, *41*(1), 34-48.

Kamel, H., Abdulah, D., & Al-Tuwaijari, J. M. (2019). Cancer classification using gaussian naive bayes algorithm. 2019 International Engineering Conference (IEC),

Kannel, W., & McGee, D. (1979). Diabetes and glucose tolerance as risk factors for cardiovascular disease: the Framingham study. *Diabetes care*, *2*(2), 120-126.

Kappert, K., Böhm, M., Schmieder, R., Schumacher, H., Teo, K., Yusuf, S., Sleight, P., & Unger, T. (2012). Impact of sex on cardiovascular outcome in patients at high cardiovascular risk: analysis of the telmisartan randomized assessment study in ACE-intolerant subjects with cardiovascular disease (TRANSCEND) and the ongoing telmisartan alone and in combination with ramipril global end point trial (ONTARGET). *Circulation*, *126*(8), 934-941.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, *30*.

Keeley, A., Hine, P., & Nsutebu, E. (2017). The recognition and management of sepsis and septic shock: a guide for non-intensivists. *Postgraduate medical journal*, *93*(1104), 626-634.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. Proceedings of ICNN'95-international conference on neural networks,

Kibria, H. B., & Matin, A. (2022). The severity prediction of the binary and multi-class cardiovascular disease− A machine learning-based fusion approach. *Computational Biology and Chemistry*, *98*, 107672.

Kilic, A. (2020). Artificial intelligence and machine learning in cardiovascular health care. *The Annals of thoracic surgery*, *109*(5), 1323-1329.

Kingma, D. P., Mohamed, S., Jimenez Rezende, D., & Welling, M. (2014). Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, *27*.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Koglin, J., Pehlivanli, S., Schwaiblmair, M., Vogeser, M., Cremer, P., & vonscheidt, W. (2001). Role of brain natriuretic peptide in risk stratification of patients with congestive heart failure. *Journal of the American College of Cardiology*, *38*(7), 1934-1941.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. Ijcai,

Krittanawong, C., Zhang, H., Wang, Z., Aydar, M., & Kitai, T. (2017). Artificial intelligence in precision cardiovascular medicine. *Journal of the American College of Cardiology*, *69*(21), 2657-2664.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84-90.

Kukar, M., Kononenko, I., Grošelj, C., Kralj, K., & Fettich, J. (1999). Analysing and improving the diagnosis of ischaemic heart disease with machine learning. *Artificial Intelligence in Medicine*, *16*(1), 25-50.

Kursa, M. B., & Rudnicki, W. R. (2011). The all relevant feature selection using random forest. *arXiv preprint arXiv:1106.5112*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444.

Lestari, W., & Sumarlinda, S. (2022). Implementation of K-Nearest Neighbor (Knn) and Suport Vector Machine (Svm) for Clasification Cardiovascular Disease. *INTERNATIONAL JOURNAL OF MULTI SCIENCE*, *2*(10), 30-36.

Li, J., Zhang, R., Shi, L., & Wang, D. (2017). Automatic whole-heart segmentation in congenital heart disease using deeply-supervised 3D FCN. International Workshop on Reconstruction and Analysis of Moving Body Organs, International Workshop on Whole-Heart and Great Vessel Segmentation from 3D Cardiovascular MRI in Congenital Heart Disease,

Loh, W. Y. (2011). Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *1*(1), 14-23.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

Luo, Z., Yetisgen-Yildiz, M., & Weng, C. (2011). Dynamic categorization of clinical research eligibility criteria by hierarchical clustering. *Journal of Biomedical Informatics*, *44*(6), 927-935.

Madani, A., Arnaout, R., Mofrad, M., & Arnaout, R. (2018). Fast and accurate view classification of echocardiograms using deep learning. *NPJ digital medicine*, *1*(1), 1-8.

Maimon, O., & Rokach, L. (2005). Data mining and knowledge discovery handbook.

Marwala, T. (2014). *Artificial intelligence techniques for rational decision making*. Springer.

Marwala, T., & Xing, B. (2018). Blockchain and artificial intelligence. *arXiv preprint arXiv:1802.04451*.

Medhekar, D. S., Bote, M. P., & Deshmukh, S. D. (2013). Heart disease prediction system using naive Bayes. *Int. J. Enhanced Res. Sci. Technol. Eng*, *2*(3).

Meikeng, Y. (2023). *INTERACTIVE: Heart disease is top cause of early deaths in Malaysia*. Retrieved 12/5/2024 from http://www.statistics.gov.my/

Mohan, S., Thirumalai, C., & Srivastava, G. (2019). Effective heart disease prediction using hybrid machine learning techniques. *IEEE Access*, *7*, 81542-81554.

Mubarak, H., Hammoudeh, A., Ahmad, S., Abdellatif, A., Mekhilef, S., Mokhlis, H., & Dupont, S. (2022). A hybrid machine learning method with explicit time encoding for improved Malaysian photovoltaic power prediction. *Journal of Cleaner Production*, 134979.

Muntasir Nishat, M., Faisal, F., Jahan Ratul, I., Al-Monsur, A., Ar-Rafi, A. M., Nasrullah, S. M., Reza, M. T., & Khan, M. R. H. (2022). A comprehensive investigation of the performances of different machine learning classifiers with SMOTE-ENN oversampling technique and hyperparameter optimization for imbalanced heart failure dataset. *Scientific Programming*, *2022*, 1-17.

Murtagh, F., & Contreras, P. (2012). Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(1), 86-97.

Nagavelli, U., Samanta, D., & Chakraborty, P. (2022). Machine learning technology-based heart disease detection models. *Journal of Healthcare Engineering*, *2022*.

Nie, F., Yuan, J., & Huang, H. (2014). Optimal mean robust principal component analysis. International conference on machine learning,

Nilashi, M., Ahmadi, H., Manaf, A. A., Rashid, T. A., Samad, S., Shahmoradi, L., Aljojo, N., & Akbari, E. (2020). Coronary heart disease diagnosis through self-organizing map and fuzzy support vector machine with incremental updates. *International Journal of Fuzzy Systems*, *22*(4), 1376-1388.

O'Meara, E., Clayton, T., McEntegart, M. B., McMurray, J. J., Piña, I. L., Granger, C. B., Östergren, J., Michelson, E. L., Solomon, S. D., & Pocock, S. (2007). Sex differences in clinical characteristics and prognosis in a broad spectrum of patients with heart failure: results of the Candesartan in Heart failure: Assessment of Reduction in Mortality and morbidity (CHARM) program. *Circulation*, *115*(24), 3111-3120.

Ozcan, M., & Peker, S. (2023). A classification and regression tree algorithm for heart disease modeling and prediction. *Healthcare Analytics*, *3*, 100130.

Pathan, M. S., Nag, A., Pathan, M. M., & Dev, S. (2022). Analyzing the impact of feature selection on the accuracy of heart disease prediction. *Healthcare Analytics*, *2*, 100060.

Ping, Y., Chen, C., Wu, L., Wang, Y., & Shu, M. (2020). Automatic detection of atrial fibrillation based on CNN-LSTM and shortcut connection. Healthcare,

Poh, M.-Z., Poh, Y. C., Chan, P.-H., Wong, C.-K., Pun, L., Leung, W. W.-C., Wong, Y.-F., Wong, M. M.-Y., Chu, D. W.-S., & Siu, C.-W. (2018). Diagnostic assessment of a deep learning system for detecting atrial fibrillation in pulse waveforms. *Heart*, *104*(23), 1921-1928.

Poplin, R., Varadarajan, A. V., Blumer, K., Liu, Y., McConnell, M. V., Corrado, G. S., Peng, L., & Webster, D. R. (2018). Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. *Nature Biomedical Engineering*, *2*(3), 158-164.

Pramanik, P. K. D., Pal, S., & Mukhopadhyay, M. (2022). Healthcare big data: A comprehensive overview. *Research Anthology on Big Data Analytics, Architectures, and Applications*, 119-147.

Puyalnithi, T., & Viswanatham, V. M. (2016). Preliminary cardiac disease risk prediction based on medical and behavioural data set using supervised machine learning techniques. *Indian J Sci Technol*, *9*(31), 1-5.

Qin, Y., Xue, L., Jiang, P., Xu, M., He, Y., Shi, S., Huang, Y., He, J., Mo, J. Q., & Guan, M. X. (2014). Mitochondrial tRNA Variants in C hinese Subjects With Coronary Heart Disease. *Journal of the American Heart Association*, *3*(1), e000437.

Quesada, J. A., Lopez-Pineda, A., Gil-Guillén, V. F., Durazo-Arvizu, R., Orozco-Beltrán, D., López-Domenech, A., & Carratalá-Munuera, C. (2019). Machine learning to predict cardiovascular risk. *International journal of clinical practice*, *73*(10), e13389.

Raina, R., Madhavan, A., & Ng, A. Y. (2009). Large-scale deep unsupervised learning using graphics processors. Proceedings of the 26th annual international conference on machine learning,

Raj, K. S., & Thinakaran, K. (2022). Prediction of Heart Disease using Forest Algorithm over K-nearest neighbors using Machine Learning with Improved Accuracy. *Cardiometry*(25), 1500-1506.

Ramesh, T., Lilhore, U. K., Poongodi, M., Simaiya, S., Kaur, A., & Hamdi, M. (2022). Predictive analysis of heart diseases with machine learning approaches. *Malaysian Journal of Computer Science*, 132-148.

Rana, J. S., Khan, S. S., Lloyd-Jones, D. M., & Sidney, S. (2021). Changes in mortality in top 10 causes of death from 2011 to 2018. *Journal of General Internal Medicine*, *36*(8), 2517-2518.

Ribas, V. J., Vellido, A., Ruiz-Rodríguez, J. C., & Rello, J. (2012). Severe sepsis mortality prediction with logistic regression over latent factors. *Expert Systems with Applications*, *39*(2), 1937-1943.

Roffo, G., Melzi, S., Castellani, U., & Vinciarelli, A. (2017). Infinite latent feature selection: A probabilistic latent graph-based ranking approach. Proceedings of the IEEE International Conference on Computer Vision,

Roffo, G., Melzi, S., Castellani, U., Vinciarelli, A., & Cristani, M. (2020). Infinite feature selection: a graph-based feature filtering approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(12), 4396-4410.

Ruiz-Fernandez, D., Torra, A. M., Soriano-Payá, A., Marin-Alonso, O., & Palencia, E. T. (2016). Aid decision algorithms to estimate the risk in congenital heart surgery. *Computer methods and programs in biomedicine*, *126*, 118-127.

Saboor, A., Usman, M., Ali, S., Samad, A., Abrar, M. F., & Ullah, N. (2022). A method for improving prediction of human heart disease using machine learning algorithms. *Mobile Information Systems*, *2022*.

Sáez, J. A., Luengo, J., Stefanowski, J., & Herrera, F. (2015). SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. *Information Sciences*, *291*, 184-203.

Sağlam, F., & Cengiz, M. A. (2022). A novel SMOTE-based resampling technique trough noise detection and the boosting procedure. *Expert Systems with Applications*, *200*, 117023.

Sainath, T. N., Kingsbury, B., Saon, G., Soltau, H., Mohamed, A.-r., Dahl, G., & Ramabhadran, B. (2015). Deep convolutional neural networks for large-scale speech tasks. *Neural Networks*, *64*, 39-48.

Savji, N., Rockman, C. B., Skolnick, A. H., Guo, Y., Adelman, M. A., Riles, T., & Berger, J. S. (2013). Association between advanced age and vascular disease in different arterial territories: a population database of over 3.6 million subjects. *Journal of the American College of Cardiology*, *61*(16), 1736-1743.

Shan, W., Qiao, Z., Heidari, A. A., Gui, W., Chen, H., Teng, Y., Liang, Y., & Lv, T. (2022). An efficient rotational direction heap-based optimization with orthogonal structure for medical diagnosis. *Computers in biology and medicine*, *146*, 105563.

Sidey-Gibbons, J. A., & Sidey-Gibbons, C. J. (2019). Machine learning in medicine: a practical introduction. *BMC medical research methodology*, *19*(1), 1-18.

Simon, T., Mary-Krause, M., Funck-Brentano, C., & Jaillon, P. (2001). Sex differences in the prognosis of congestive heart failure: results from the Cardiac Insufficiency Bisoprolol Study (CIBIS II). *Circulation*, *103*(3), 375-380.

Sitar-tăut, A., Zdrenghea, D., Pop, D., & Sitar-tăut, D. (2009). Using machine learning algorithms in cardiovascular disease risk evaluation. *Age*, *1*(4), 4.

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *arXiv preprint arXiv:1206.2944*.

Sohn, K., Lee, H., & Yan, X. (2015). Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, *28*.

Son, J., Shin, J. Y., Chun, E. J., Jung, K.-H., Park, K. H., & Park, S. J. (2020). Predicting high coronary artery calcium score from retinal fundus images with deep learning algorithms. *Translational vision science & technology*, *9*(2), 28-28.

Sowmiya, C., & Sumitra, P. (2021). A hybrid approach for mortality prediction for heart patients using ACO-HKNN. *Journal of Ambient Intelligence and Humanized Computing*, *12*(5), 5405-5412.

Stanek, B., Frey, B., Hülsmann, M., Berger, R., Sturm, B., Strametz-Juranek, J., Bergler-Klein, J., Moser, P., Bojic, A., & Hartter, E. (2001). Prognostic evaluation of neurohumoral plasma levels before and during beta-blocker therapy in advanced left ventricular dysfunction. *Journal of the American College of Cardiology*, *38*(2), 436-442.

Statlog (Heart) Data Set. UCI Machine Learning Repositry. Retrieved 21/10/2021 from https://archive.ics.uci.edu/ml/datasets/statlog+(heart)

Sun, J., Yang, Y., Wang, Y., Wang, L., Song, X., & Zhao, X. (2020). Survival Risk Prediction of Esophageal Cancer Based on Self-Organizing Maps Clustering and Support Vector Machine Ensembles. *IEEE Access*, *8*, 131449-131460. https://doi.org/10.1109/ACCESS.2020.3007785

Tama, B. A., Im, S., & Lee, S. (2020). Improving an intelligent detection system for coronary heart disease using a two-tier classifier ensemble. *BioMed Research International*, *2020*.

Tan, L. K., Liew, Y. M., Lim, E., & McLaughlin, R. A. (2017). Convolutional neural network regression for short-axis left ventricle segmentation in cardiac cine MR sequences. *Medical image analysis*, *39*, 78-86.

Taylor, C. J., Ordóñez-Mena, J. M., Roalfe, A. K., Lay-Flurrie, S., Jones, N. R., Marshall, T., & Hobbs, F. R. (2019). Trends in survival after a diagnosis of heart failure in the United Kingdom 2000-2017: population based cohort study. *bmj*, *364*.

Thanga Selvi, R., & Muthulakshmi, I. (2021). An optimal artificial neural network based big data application for heart disease diagnosis and classification model. *Journal of Ambient Intelligence and Humanized Computing*, *12*(6), 6129-6139.

Thompson, K., Venkatesh, B., & Finfer, S. (2019). Sepsis and septic shock: current approaches to management. *Internal medicine journal*, *49*(2), 160-170.

Tirosh, A., Shai, I., Afek, A., Dubnov-Raz, G., Ayalon, N., Gordon, B., Derazne, E., Tzur, D., Shamis, A., & Vinker, S. (2011). Adolescent BMI trajectory and risk of diabetes versus coronary disease. *New England Journal of Medicine*, *364*(14), 1315-1325.

Tison, G. H., Sanchez, J. M., Ballinger, B., Singh, A., Olgin, J. E., Pletcher, M. J., Vittinghoff, E., Lee, E. S., Fan, S. M., & Gladstone, R. A. (2018). Passive detection of atrial fibrillation using a commercially available smartwatch. *JAMA cardiology*, *3*(5), 409-416.

Tiwari, A., Chugh, A., & Sharma, A. (2022). Ensemble framework for cardiovascular disease prediction. *Computers in biology and medicine*, 105624.

Topîrceanu, A., & Grosseck, G. (2017). Decision tree learning used for the classification of student archetypes in online courses. *Procedia Computer Science*, *112*, 51-60.

Tortajada, S., Robles, M., & García-Gómez, J. M. (2015). Incremental logistic regression for customizing automatic diagnostic models. In *Data Mining in Clinical Medicine* (pp. 57-78). Springer.

Tsao, C. W., Aday, A. W., Almarzooq, Z. I., Alonso, A., Beaton, A. Z., Bittencourt, M. S., Boehme, A. K., Buxton, A. E., Carson, A. P., & Commodore-Mensah, Y. (2022). Heart Disease and Stroke Statistics—2022 Update: A Report From the American Heart Association. *Circulation*, *145*(8), e153-e639.

Tunstall-Pedoe, H., Kuulasmaa, K., Mähönen, M., Tolonen, H., Ruokokoski, E., & Amouyel, P. (1999). Contribution of trends in survival and coronar y-event rates to changes in coronary heart disease mortality: 10-year results from 37 WHO MONICA Project populations. *The Lancet*, *353*(9164), 1547-1557.

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC medical informatics and decision making*, *19*(1), 1-16.

Umer, M., Sadiq, S., Karamti, H., Karamti, W., Majeed, R., & Nappi, M. (2022). IoT Based Smart Monitoring of Patients' with Acute Heart Failure. *Sensors*, *22*(7), 2431.

Van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine learning*, *109*(2), 373-440.

Vembandasamy, K., Sasipriya, R., & Deepa, E. (2015). Heart diseases detection using Naive Bayes algorithm. *International Journal of Innovative Science, Engineering & Technology*, *2*(9), 441-444.

Vivekanandan, T., & Narayanan, S. J. (2019). A hybrid risk assessment model for cardiovascular disease using cox regression analysis and a 2-means clustering algorithm. *Computers in biology and medicine*, *113*, 103400.

Wadhonkar, B., Tijare, P., & Sawalkar, S. (2015). A data mining approach for classification of heart disease dataset using neural network. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, *4*(5), 426-433.

Wang, X., & Paliwal, K. K. (2003). Feature extraction and dimensionality reduction algorithms and their applications in vowel recognition. *Pattern Recognition*, *36*(10), 2429-2439.

Wang, Y., Simon, M. A., Bonde, P., Harris, B. U., Teuteberg, J. J., Kormos, R. L., & Antaki, J. F. (2012). Decision tree for adjuvant right ventricular support in patients receiving a left ventricular assist device. *The Journal of heart and lung transplantation*, *31*(2), 140-149.

Waqar, M., Dawood, H., Dawood, H., Majeed, N., Banjar, A., & Alharbey, R. (2021). An efficient smote-based deep learning model for heart attack prediction. *Scientific Programming*, *2021*.

Weng, S. F., Reps, J., Kai, J., Garibaldi, J. M., & Qureshi, N. (2017). Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PLoS ONE*, *12*(4), e0174944.

Xia, Y., Wulan, N., Wang, K., & Zhang, H. (2018). Detecting atrial fibrillation by deep convolutional neural networks. *Computers in biology and medicine*, *93*, 84-92.

Xiong, Z., Nash, M. P., Cheng, E., Fedorov, V. V., Stiles, M. K., & Zhao, J. (2018). ECG signal classification for the detection of cardiac arrhythmias using a convolutional recurrent neural network. *Physiological measurement*, *39*(9), 094006.

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295-316.

Yuvalı, M., Yaman, B., & Tosun, Ö. (2022). Classification Comparison of Machine Learning Algorithms Using Two Independent CAD Datasets. *Mathematics*, *10*(3), 311.

Zaragoza, C., Gomez-Guerrero, C., Martin-Ventura, J. L., Blanco-Colio, L., Lavin, B., Mallavia, B., Tarin, C., Mas, S., Ortiz, A., & Egido, J. (2011). Animal models of cardiovascular diseases. *Journal of Biomedicine and Biotechnology*, *2011*.

Zhao, Z., Särkkä, S., & Rad, A. B. (2018). Spectro-temporal ECG analysis for atrial fibrillation detection. 2018 IEEE 28Th international workshop on machine learning for signal processing (MLSP),

Zhu, C., Idemudia, C. U., & Feng, W. (2019). Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques. *Informatics in Medicine Unlocked*, *17*, 100179.