Survival Analysis and Machine Learning Models for Predicting Heart Failure Outcomes

Naseem Mohammed ALQahtani[1], Abdulmohsen Algarni[2]
Department of Informatics and Computer Systems-College of Computer Science,
King Khalid University, Abha 61421, Saudi Arabia[1]
Department of Computer Science, King Khalid University, Abha 61421, Saudi Arabia[2]

***Abstract*—Heart failure is still one of the prominent causes of morbidity and mortality globally, and thus, determining the principal factors influencing survival in patients becomes crucial. Being able to predict survival is critical for optimizing patient treatment and management. Heart failure, with its multifactorial and involvement of numerous clinical variables, complicates prediction of survival rates in patients. This study utilizes the "Heart Failure Clinical Records" dataset to analyze and predict patient survival based on two separate approaches: survival analysis and machine learning (ML) classification. Specifically, we employ the Cox Proportional Hazards Model to assess the influence of clinical variables like "age", "serum creatinine", and "ejection fraction" on survival durations. Additionally, machine learning classification models like K-Nearest Neighbors (KNN), Decision Trees (DT), and Random Forests (RF) are implemented to predict the binary response variable of survival (DEATH_EVENT). Data preprocessing is carried out using methods like feature scaling, imputation of missing values, and balancing the classes for the improvement of model performance. Among the evaluated models, the Random Forest classifier, when integrated with feature selection derived from the Cox model, reached the best performance with 96.2% accuracy and an AUC ROC of 0.987, outperforming all other approaches. The results indicate that integrating survival analysis with machine-learning techniques is effective in heart failure prediction outcomes, providing valuable support for patient management and clinical decision-making.**

***Keywords*—Heart failure prediction; machine learning; cox proportional hazards model; random forest**

I. Introduction

The heart is considered to be among the most vital components within the human body, which is responsible for the essential task of circulating blood throughout the entire system. Heart disease (HD) is a medical condition that adversely impacts the heart's proper functioning. It encompasses various forms, like heart failure and coronary artery disease (CAD), which is a prevalent type of heart ailment. The primary culprit behind CAD is the constriction or obstruction of the coronary blood vessels [1]. In recent years, cardiovascular or heart disease has consistently held the dubious distinction of being the main cause of mortality globally. As per approximations from the World Health Organization (WHO), there could be around 17.9 million fatalities related to heart issues yearly, with CAD and cerebral strokes jointly contributing to 80% of these deaths [2]. HD can result from an array of risk factors, including genetic predispositions, personal and professional behaviors, and lifestyle choices. Therefore, an early, accurate medical assessment for heart disease is crucial in implementing preventive measures to reduce mortality rates [3].

Early detection or diagnosis of heart disease is necessary because it may be a significant problem. Different methods are used to diagnose heart disease. Angiography is a method that is becoming more popular among doctors. However, there were some drawbacks to the angiography method, such as the expensive process that was used and the requirements that doctors needs, to examine multiple factors in order to diagnose a disease. As a result, this procedure can be extremely hard on doctors, and these drawbacks have prompted researchers to develop non-invasive methods to predict heart problems. The medical reports of patients can be handled by conservative medical approaches. These cautious approaches are carried out by humans, which could make them time-consuming and lead to inaccurate results [4].

In today's digital age, the fast evolution in the areas of science and technology results in the production of huge volumes of healthcare data utilizing diverse technologies such as embedded systems, intelligent health devices, and computers, which have become more popular due to the rapid development in these fields. Machine learning algorithms are progressively being envisioned as effective agents within the healthcare industry, where they can effectively be utilized to diagnose and forecast diseases in advance according to recognizing significant patterns in the data [5].

This study contributes to the field of heart failure survival prediction through the application of a two-framework approach that combines survival analysis and machine learning techniques. It compares a number of ML techniques to determine the best approach for prediction of patient's outcome. Furthermore, real-world applications of these predictive models are emphasized, illustrating their potential utility in the clinic to improve treatment decision-making and patient outcomes. In this study, three models were employed, and the RF model achieved the highest accuracy of 96.21%, also outperforming all related works in terms of predictive performance.

This study seeks to answer the central research question: Can the integration of survival analysis and machine learning techniques enhance the prediction of survival outcomes in heart failure patients compared to existing methods? This question frames the comparative analysis and drives the evaluation of clinical relevance and model performance.

In the following sections, Section II presents related work in survival prediction using machine learning. Section III presents the suggested approach to combining survival analysis and machine learning. Section IV illustrates the results, the model performance, and explains the findings and their clinical applicability. In Section V, a conclusion for the study and future work suggestions are presented.

II. Related Work

A number of research studies have been conducted on using statistical models and ML in survival prediction in patients with heart failure. Various techniques such as SMOTE, Random Forests, and Cox Proportional Hazards have been employed in order to achieve higher accuracy predictions and mitigate the issues of class imbalance. Such models have indicated great potential for optimizing clinical decision-making and patient care by discovering risk factors as well as optimization of survival prediction.

For instance, Ishaq et al. utilized the Synthetic Minority Oversampling Technique (SMOTE) along with other data mining techniques to optimize survival rates' predictive accuracy among heart failure patients. Their comparative study of nine machine learning models revealed that the Extra Tree Classifier (ETC), when paired with SMOTE, achieved the highest accuracy of 92.62%, underscoring the value of handling imbalanced data [6]. Rahayu et al. wanted to decrease heart failure mortality rates by using ML classifiers on the "Heart Failure Clinical Records" dataset. They experimented with different models like RF, DT, KNN, SVM, ANN, and NB. The RF model with resampling yielded the best accuracy of 94.31% that was a bit better than Ishaq et al. This suggests that ensemble techniques could be immensely helpful in clinical prediction [7]. Furthermore, Oladimeji et al. enhanced prediction accuracy by incorporating feature selection and class balancing into their machine learning. Their findings determined that “age”, “smoking status”, “serum creatinine”, and “ejection fraction” are critical variables for predicting survival, which demonstrates how crucial it is to include the pertinent clinical features in the input to the model [8]. In addition, Lee et al. combined Kaplan-Meier survival curves alongside Cox regression modeling on the same data. "Age", "serum creatinine", and "ejection fraction" were found to be important predictors of mortality in their study, demonstrating the strength of merging statistical and machine learning methodologies in biomedical informatics [9]. Using these findings, Mamun et al. employed models like Logistic Regression, XGBoost, and LightGBM to predict survival from heart failure. LightGBM surpassed other models with 85% accuracy and a 93% AUC score, yet again establishing the feasibility of ML in predicting high-risk patients [10]. The key information for each related work is summarized in Table I.

TABLE I. Summary of Related Work

Writer	Paper	Year	Models	Dataset	Results
Ishaq et al. [6]	“Improving the Prediction of Heart Failure Patients Survival Using SMOTE and Effective Data Mining Techniques”	2021	DT, AB, LR, SGD, RF, GBM, ETC, GNB, SVM	UCI 299-patient HF clinical records	Extra Tree Classifier + SMOTE achieved 92.62% accuracy.
Rahayu et al. [7]	“Prediction Of Survival Of Heart Failure Patients Using Random Forest”	2020	RF, DT, KNN, SVM, ANN, NB	UCI 299-patient HF clinical records	Random Forest + Resampling achieved 94.31% accuracy. Resampling outperformed SMOTE (85.82%).
Oladimeji et al. [8]	“Predicting Survival of Heart Failure Patients Using Classification Algorithms”	2020	KNN, SVM, NB, RF	UCI 299-patient HF clinical records	Random Forest achieved 83.17% accuracy.
Lee et al. [9]	“Machine Learning-Enhanced Survival Analysis: Identifying Significant Predictors of Mortality in Heart Failure”	2024	CoxPH, KM	UCI 299-patient HF clinical records	C-index = 0.77.
Mamun et al. [10]	“Heart failure survival prediction using machine learning algorithm: am I safe from heart failure?”	2022	LR, DT, SVM, XGB, LGBM, RF, KNN, BAG	UCI 299-patient HF clinical records	LightGBM yielded 85 % accuracy, AUC 93 %

Earlier studies have advanced heart-failure survival modelling, but each leaves critical gaps that our work will bridge. Ishaq et al. emphasised class-imbalance handling and tried nine classifiers, yet they depended on a single oversampling method and a coarse Random-Forest ranking that can blur clinically meaningful variables [6]. Rahayu et al. explored resampling but limited themselves to the original 299-patient cohort and ignored any time-to-event analysis, making their findings hard to translate into bedside risk estimates [7]. Oladimeji et al. improved Weka-based models with heuristic feature rankings, though their single hybrid sampler and scant justification for the chosen variables weaken the model’s clinical defensibility [8]. Our study will couple Cox-based hazard significance with multiple balancing strategies and scaling pipelines, producing a compact, interpretable feature core and a classifier that remains stable across richer, more realistic data landscapes.

More recent work revisits the same UCI cohort with modern tools but still stops short of an integrated survival-ML framework. Lee et al. rely solely on Cox regression, leaving unexplored how ensemble learners might amplify discrimination or how larger cohorts shift variable importance, while Mamun et al. benchmark eight off-the-shelf classifiers and highlight LightGBM without survival-specific metrics or built-in explainability [9] [10]. Our study will extend classical survival statistics into an ensemble pipeline, embed explainability through hazard-filtered features and calibrated probability curves, and validate performance on an expanded, balanced dataset. By unifying statistical survival analysis with machine-learning robustness and interpretability, our work will deliver insights that are both clinically actionable and generalisable—advancing the field beyond retrospective accuracy contests towards real-world decision support.

III. METHODOLOGY

In this study, an overall approach was provided to model and examine heart failure survival data using a well-defined multi-stage process. First the data from Kaggle was imported, and the dataset contains clinical records of heart failure patients. The initial step in the preprocessing stage was to handle duplicate records in such a way that all records in the data are single case records to ensure the integrity of the analysis. Having preprocessed the data, the Cox Proportional Hazard Model was utilized in performing the survival analysis. This enabled us to model how various clinical features are associated with the patients' survival time. Through this, the key traits that govern the survival of a patient are revealed. The output of the feature analysis of the Cox model was then used to perform feature selection, retaining only the variables that were found to have a great impact on survival. This step streamlined the dataset and ensured that only relevant features were used in subsequent modeling, which improves both accuracy and interpretability.

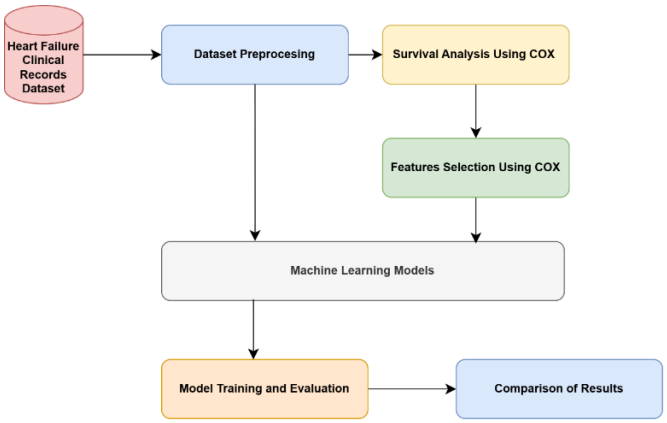

Fig. 1. The proposed methodology.

Fig. 1 illustrates the proposed methodology for analyzing and predicting patient survival in heart failure. The process of the proposed framework is as follows:

- Dataset Acquisition: Obtain Heart Failure Clinical Records dataset.
- Data Preprocessing: Clean the dataset through eliminating duplicates and handling missing values.
- Survival Analysis: Apply Cox Proportional Hazards Model to identify significant clinical features.
- Feature Selection: Select key predictors based on Cox model significance.
- Machine Learning Modeling: Apply classification algorithms (DT, KNN, RF).
- Model Training and Evaluation: Train models and evaluate using Accuracy, F1-score, AUC-ROC, Precision, and Recall.
- Results Comparison: Identify the best-performing predictive model.

Compared to prior frameworks, our approach offers several advantages: 1) it combines time-to-event modeling with ensemble learning for a more nuanced prediction of survival; 2) it uses clinically interpretable feature selection through Cox regression, improving transparency; and 3) it systematically integrates multiple class balancing techniques and feature scaling methods to enhance robustness. These design choices make the framework more adaptable to real-world clinical data than models that use only classification algorithms or only statistical survival analysis.

A. Survival Analysis Using COX

The Cox proportional hazards model (also called Cox regression, CoxPH, or Cox's model) has been the most commonly employed method for examining the association between a patient's survival and potential risk factors which is known as survival analysis [11]. The h_i value depends on the predictor variables (x) and baseline hazard function h0. A convenient feature of this modelling method is that the baseline hazard function h0 does not need to be explicitly modelled or estimated, and the modelling task involves only estimating the β parameters for the effects of predictor x. In simpler terms, the baseline hazard function doesn't rely on any specific assumptions, and the predictors x multiply the hazard proportionally through an exponential function (for instance, the below Eq. (1) provides an example using two predictors):

$$h_i(t) = h_0(t)e^{\beta_1 * x_1 + \beta_2 * x_2} \quad (1)$$

The Cox model, which assumes constant hazard ratios over time, was used on the data to predict mortality. In this study, the lifelines library was used to fit the CoxPH model to the "Heart Failure Clinical Records" dataset, with "time" as the duration column (survival time) and "DEATH_EVENT" as the event indicator (death occurrence). The model was fitted using the Breslow method for estimating the baseline hazard since this works best when survival times are tied. The model included eleven clinical variables as predictors: "age", "anemia", "creatinine phosphokinase", "diabetes", "ejection fraction", "high blood pressure", "platelets", "serum creatinine", "serum sodium", "sex", and "smoking status". Analysis proved that "age", "ejection fraction", "serum creatinine", and "high blood pressure" were statistically significant predictors for survival ($p < 0.005$), meaning that they were highly correlated with mortality risk. The model achieved a concordance index (C-index) of 0.76, which indicates that it is highly capable of discriminating between surviving patients and non-survivors. The log-likelihood ratio test yielded a statistically significant result ($\chi^2 = 347.20$, $p < 0.005$), further confirming the model's explanatory power.

In order to visualize how each binary clinical feature affects survival, a number of survival curves were drawn for six of the most significant covariates found through analysis: sex, high blood pressure, anemia, smoking, diabetes, and serum creatinine. Each plot compares the survival probabilities between patients with (marked in blue) and without (marked in orange) the condition over time. Fig. 2 below provides a clearer interpretation of how each factor contributes to overall mortality risk in heart failure patients.

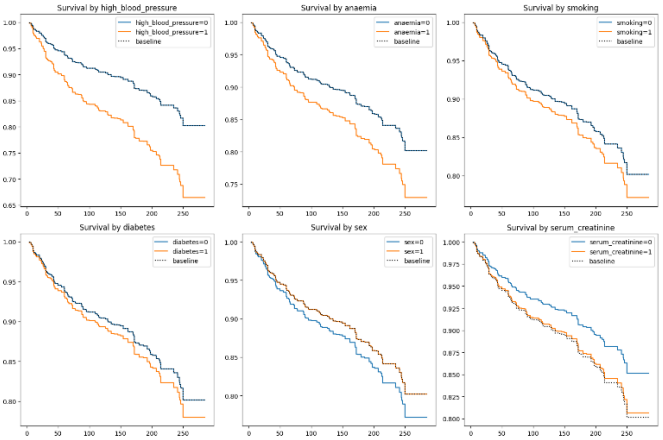

Fig. 2. Comparative survival curves highlighting the impact of clinical variables on heart failure patient outcomes.

Observations from the plots are:

- High Blood Pressure: Patients with high blood pressure (value=1) exhibit notably lower survival probabilities compared to those without it, indicating a significant negative impact on survival.
- Anemia: Presence of anemia slightly decreases patient survival probabilities, suggesting it moderately affects mortality risk.
- Smoking: Smoking status shows minimal differences between groups, implying a smaller impact than anticipated.
- Diabetes: Survival curves for patients with and without diabetes are similar, suggesting that diabetes has a limited effect on survival probability.
- Sex: There is minimal difference in survival probabilities based on sex, suggesting that gender alone has limited predictive power.
- Serum Creatinine: Higher serum creatinine levels (value=1) are clearly related with reduced survival probabilities, underscoring its importance as a predictor of mortality risk.

Fig. 3 presents the baseline hazard function, which provides critical context for interpreting the results of the Cox model. It represents the time-dependent risk of death for a hypothetical patient with average or baseline values for all covariates. Visualizing this function helps reveal how the risk of death evolves over the follow-up period, independent of individual patient characteristics.

The baseline survival function serves as a counterpart to the baseline hazard function, depicting the likelihood of survival over time for a reference patient—defined as an individual whose covariates are all set to their baseline or standard values. Fig. 4 represents the estimated baseline survival function over time from the CoxPH model.

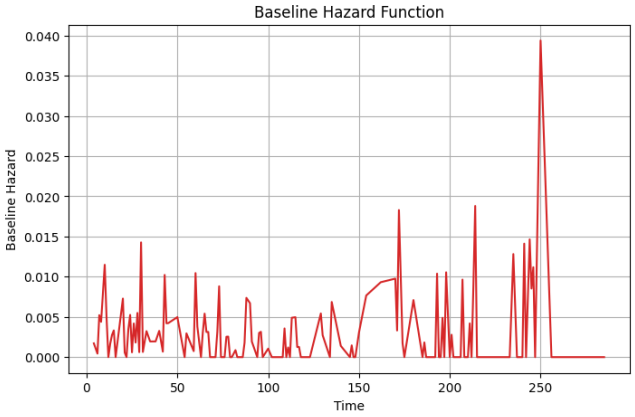

Fig. 3. Estimated baseline hazard function over time from the CoxPH model.

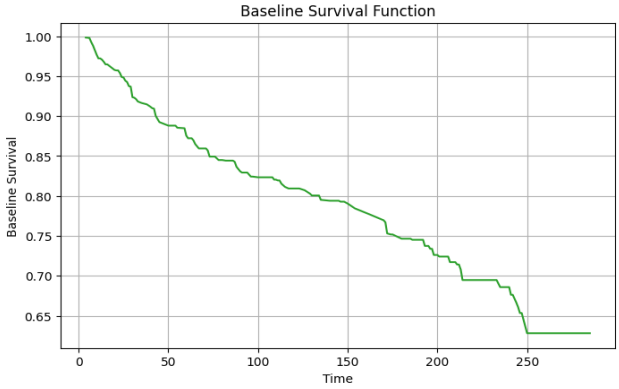

Fig. 4. Estimated baseline survival function over time from the CoxPH model.

B. Feature Selection Using COX

In this study, the Cox Model was not only used to analyze survival times but also employed as a feature selection tool. After fitting the Cox model, the statistical significance of each covariate was evaluated using the p-value associated with its coefficient. Covariates with p-values less than 0.005 were considered statistically significant as shown in Table II and were selected for downstream machine learning classification tasks. The Cox library in Python defaults to a significance level of 0.005. Notably, no variables were found with P-values between 0.005 and 0.05, indicating that the excluded variables had P-values greater than 0.05. The seven selected features include: “age”, “anemia”, “creatinine phosphokinase”, “ejection fraction”, “high blood pressure”, “serum creatinine”, “serum sodium”. These variables demonstrated strong associations with patient survival, indicating their clinical relevance to predict death risk in heart failure patients. By limiting the ML inputs to these statistically significant features, the classification models could focus on the most informative predictors, reducing noise from irrelevant variables and improving both predictive accuracy and model interpretability.

Table II summarizes the results of Cox model for dataset features.

TABLE II. SUMMARY OF COXPH MODEL RESULTS FOR DATASET FEATURES

Feature	Coef	exp(Coef)	SE(Coef)	Coef 95% CI (Lower)	Coef 95% CI (Upper)	exp(Coef) 95% CI (Lower)	exp(Coef) 95% CI (Upper)	Z-score	p-value	-log2(p)
Age	0.04	1.04	0.00	0.03	0.05	1.03	1.05	8.60	<0.005	56.85
Anemia	0.36	1.43	0.10	0.15	0.56	1.16	1.75	3.42	<0.005	10.63
Creatinine Phosphokinase	0.00	1.00	0.00	0.00	0.00	1.00	1.00	4.16	<0.005	14.96
Diabetes	0.12	1.12	0.11	-0.09	0.33	0.91	1.39	1.09	0.27	1.86
Ejection Fraction	-0.05	0.95	0.01	-0.06	-0.04	0.94	0.96	-9.30	<0.005	65.93
High Blood Pressure	0.62	1.85	0.10	0.41	0.82	1.51	2.27	5.92	<0.005	28.23
Platelets	-0.00	1.00	0.00	-0.00	0.00	1.00	1.00	-1.01	0.31	1.67
Serum Creatinine	0.29	1.33	0.04	0.22	0.36	1.24	1.43	8.06	<0.005	50.26
Serum Sodium	-0.05	0.95	0.01	-0.08	-0.03	0.93	0.97	-5.00	<0.005	20.74
Sex	-0.16	0.85	0.12	-0.40	0.08	0.67	1.08	-1.30	0.19	2.38
Smoking	0.16	1.17	0.12	-0.08	0.40	0.93	1.49	1.32	0.19	2.43

C. Data Balancing Techniques

1) Random Over-Sampling. Random Over-Sampling balances binary classification datasets by replicating original samples, thereby increasing the dataset size without creating new types of samples. It handles both continuous and categorical data but doesn't introduce new variations [12].

2) SMOTE. SMOTE addresses class imbalance by creating synthetic minority-class instances based on nearest neighbors using Euclidean distance. Although effective, it can introduce extra noise, particularly in high-dimensional data [13].

3) Random Under-Sampling. Random Under-Sampling balances class distribution by randomly removing examples from the majority class, simplifying dataset size and addressing imbalance effectively [14].

D. Machine Learning Models

1) Decision tree. A DT is a tree-model, where every node is a split of the data based on some features, the branches are the results of the splits, and the leaves are the final classifications. Prior to the construction of a DT, the most discriminative feature for accurate classification must be found. This supervised learning approach works by recursively dividing the data into smaller-sized subsets, based on input variable values, until certain stopping conditions are fulfilled [15]. Therefore, it is important to set a feature assessment criterion. In a DT, the “setting” criterion defines how the tree nodes are to be divided and the “log_loss” criterion is aimed at log loss minimization, i.e., a measure of misclassification errors. Model complexity is defined in parallel by the "max_depth" parameter, which regulates how deeply the tree can grow [15]. In this study, Grid Search was used to enhance the hyperparameters of the DT, including “max_depth” (values: 3, 5, 10, 15, 20) and “min_samples_split” (values: 2, 5, 10), to achieve the optimum performance.

2) Random forest. The Random Forest (RF) technique is widely used to solve classification and regression issues. It predicts by integrating a series of hierarchical, tree-like decision models. The method is very suitable for generating consistent outcomes, even when a lot of the data has missing values [16]. The Decision Tree samples can be utilized as additional data. RF is an ensemble learning method which combines many DTs with the aim of getting a more precise solution to prediction issues. It is supervised learning with enhanced general performance by putting Decision Tree concepts into practice [16]. Two steps involve the application of the RF approach. In step one, a DT is constructed. In step two, prediction is made by a first-stage tree classifier. Complexity is affected under the control of the individual tree depths through the "max_depth" parameter. The “min_samples_split” prevents overfitting by determining the number of samples for splitting a node, as DT does. The "n_estimators" parameter specifies the number of DTs to use [16]. In this research study, Grid Search was employed in optimizing the Random Forest parameters, i.e., "max_depth" (values: 10, 20, 30, 40, 50), "min_samples_split" (values: 2, 5, 10), and "n_estimators" (values: 50, 100, 150, 200), to enhance the prediction accuracy of the model.

3) KNN. The KNN is an extremely critical grading tool that utilizes available data set information in order to categorize new instances of data [17]. It is unique in that it prioritizes keeping the entire dataset rather than adding previously learnt information. In order to classify new points, the KNN uses the feature space's nearest neighbor's class labels [17]. It uses the Euclidean distance approach in order to establish the closeness of points with respect to the newly encountered point. The distance between points in the training dataset and the new point is utilized in order to give scores, with a unity score given to the k-point in the smallest gap. The number of closest neighbors computed, referred to by the term K, is a hyperparameter that must be adjusted according to the type of data and the specific context in focus. The kNN approach can be described as: Choosing a parameter k among the nearby points is step one. Finding the Euclidean distance among the k nearest neighbors chosen is step two. Calculating the KNN using Euclidean distance is step three. Counting the points in

each class among the k nearest neighbors is step four. In the fifth, new points are assigned to the most surrounding neighboring categories. It is the model construction completion process. Choosing the value of k in the KNN algorithm effectively controls how the model trades off between the bias and variance [17]. With a very small k, like 1 or 3, there will be too much variance, i.e., the model will overfit the training set by learning the noise and won't generalize well to new data. A large k, however, is likely to yield a model with too much bias, which will do badly by not fitting the training set very well. The 'algorithm' parameter defines which algorithm is used by the KNN model, and "ball tree" is a fast and effective option. The "leaf_size" parameter sets the number of data points stored in each leaf node of the tree, and the "metric" option specifies the way that the distance is calculated. The "weights" option influences the weighting of predictions based on the neighbors' contribution [17]. Grid Search was utilized for the optimization of the hyperparameters of the KNN model such as "n_neighbors" (values: 3, 5, 7, 9, 11, 15, 21, 25) and "leaf_size" (values: 10, 20, 30, 40, 50) in this study for best-in-class classification.

4) Evaluation metrics. The model performances were evaluated against certain key metrics: confusion matrix, precision, recall, accuracy, and F1-score. A confusion matrix is a table representation of the format in which predicted results are compared with actual values split into four parts [18].

In classification tasks:

- True Positive (TP): the model correctly identifies an instance that is actually positive.
- True Negative (TN): the model correctly identifies an instance that is actually negative.
- False Positive (FP): the model incorrectly labels a negative instance as positive.
- False Negative (FN): the model incorrectly labels a positive instance as negative.

These components help in understanding the classification performance in more detail. The structure of the confusion matrix is illustrated in Fig. 5.

- Accuracy: Accuracy represents the proportion of correct predictions made by the model out of all predictions performed, where it shows us, in a straightforward way, how often the model gets things right [18] [see Eq. (2)].

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

- Precision: Precision (often called "positive predictive value") measures how reliable the model's positive predictions are. In other words, it tells us what fraction of the cases the model flags as positive are truly positive. [18] [see Eq. (3)].

$$\text{Precision} = \frac{TP}{TP+FP} \quad (3)$$

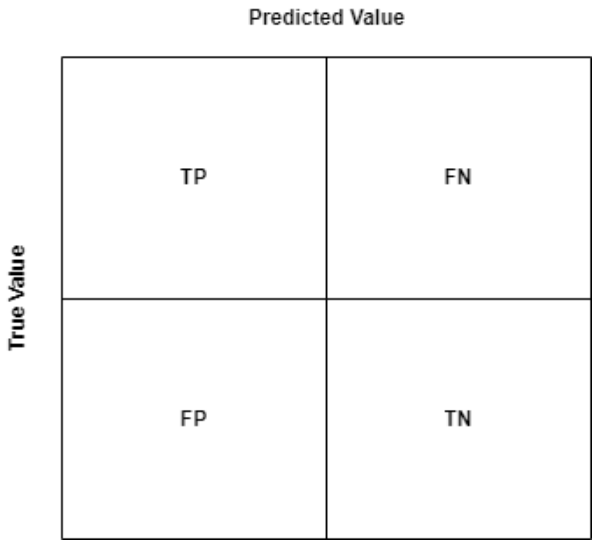

Fig. 5. Confusion matrix.

- Recall: Recall measures the capacity of model to capture all the true positive cases, such that, it's the ratio of properly detected positive instances out of all actual positive instances [18] [see Eq. (4)].

$$\text{Recall} = \frac{TP}{TP+FN} \quad (4)$$

- F1-Score: The F1-score blends precision and recall into one metric, yielding a single value that captures both aspects [18] [see Eq. (5)].

$$F1 - \text{score} = \frac{2\times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (5)$$

IV. Results

This section provides an in-depth analysis of the results obtained through the application of feature selection using the Cox proportional hazards model, combined with various preprocessing. In this study, the performance of three classifiers, KNN, DT, and RF, was evaluated on our dataset. Our analysis begins with define dataset, and moves to classifiers' performance on data without feature selection, without features scaling, and progresses to results incorporating Cox-selected features and different data balancing methods.

A. Dataset

In this study, the "Heart Failure Clinical Records Dataset" available on Kaggle was utilized [19]. This dataset is an extended version of the original dataset from the "UCI Machine Learning Repository" [20], which contained medical records of 299 patients with heart failure. The Kaggle version expands the dataset to include 5000 patient records, each with thirteen clinical features obtained during the follow-up period. The original dataset was used in the reference [21], where it demonstrated the potential of ML in predicting patient survival based on key clinical features like "ejection fraction" and "serum creatinine".

The dataset includes the below columns as shown in Table III.

TABLE III. DATASET FEATURES

Column Name	Description	Unit
Age	Patient's age	Years
Anemia	Presence of reduced red blood cells or hemoglobin	Boolean
Creatinine Phosphokinase (CPK)	CPK enzyme level in blood	mcg/L
Diabetes	Whether the patient has diabetes	Boolean
Ejection Fraction	Blood percentage pumped out of the heart each time it contracts	Percentage
High Blood Pressure	Whether the patient has high blood pressure	Boolean
Platelets	Platelet count in blood	kiloplatelets/mL
Sex	Male or Female	Binary
Serum Creatinine	Creatinine level in blood	mg/dL
Serum Sodium	Sodium level in blood	mEq/L
Smoking	Whether the patient smokes	Boolean
Time	Duration of follow-up	Days
DEATH_EVENT	Whether the patient passed away during the follow-up	Boolean

B. Dataset Preprocessing

The initial step in preprocessing was to check the dataset for missing values and ensure that there were no null values. Duplicate rows were present in the dataset, nevertheless, and were eliminated to preserve data quality and prevent biased learning. Following cleaning, a number of scaling techniques were used to compare them. Specifically, the following methods were applied:

1) Standard scaler: This technique standardizes the data by transforming it to have a mean of zero and a standard deviation of one. This helps all features contribute proportionally, particularly when they do not have all the same units. The equation is Eq. (6):

$$scaled_x = \frac{x-\mu}{\sigma} \quad (6)$$

Such that x represents the original value, μ represents the mean, and σ stands for the standard deviation.

2) Min-max scaler: It rescales each feature to a fixed range, generally [0, 1], but preserves the shape of the distribution while altering the scale. The equation is Eq. (7):

$$scaled_x = \frac{x-MinX}{MaxX-MinX} \quad (7)$$

where, x is the value, $minx$ is the minimum, and $maxx$ is the maximum.

3) MaxAbs scaler: This technique divides each value by the feature's maximum absolute value, scaling to the range [1,1-]. The equation is Eq. (8):

$$scaled_x = \frac{x}{maxX\vee} \quad (8)$$

Fig. 6 represents a horizontal bar plot to show class distribution after dropping all duplicated rows. It demonstrates a noticeable class imbalance, where the number of patients who survived (class 0) significantly exceeds the number of patients who experienced a death event (class 1). This imbalance necessitates the use of specialized data balancing techniques to ensure unbiased and reliable predictions by machine learning models.

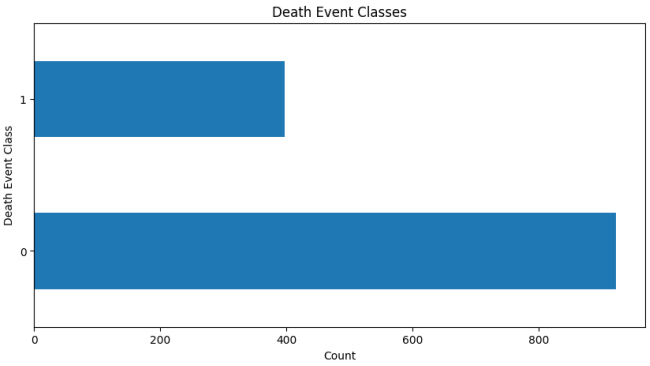

Fig. 6. Class distribution in dataset.

C. Model Optimization and Scaling Analysis

To maximize model performance, hyperparameters were tuned using Grid Search and evaluated three scaling methods: StandardScaler, MinMaxScaler, and MaxAbsScaler. Table IV summarizes the optimal hyperparameters after grid search.

TABLE IV. OPTIMAL HYPERPARAMETERS AFTER GRID SEARCH

Model	Hyperparameter	Optimal Value	Explanation
DT	max_depth	10	Restricts the depth of the tree to prevent overfitting, while still capturing meaningful patterns in the data.
	min_samples_split	2	Ensures nodes split even with minimal samples, improving granularity.
KNN	n_neighbors	5	Balances sensitivity (smaller k) and noise resistance (larger k).
	leaf_size	10	Optimizes query speed vs. accuracy in nearest neighbor searches.
RF	max_depth	10	Controls individual tree complexity for better generalization.
	min_samples_split	2	Similar to DT, ensures finer splits for imbalanced data.
	n_estimators	1000	More trees increase robustness at the cost of computational expense.

Table V represents a comparison of the scaling methods for DT, KNN, and RF.

Decision Trees and Random Forests, both tree-based models, are naturally unaffected by feature scaling since they split data based on thresholds rather than distances. In contrast, K-Nearest Neighbors (KNN) depends on distance calculations, and its performance improved noticeably with StandardScaler, reaching an accuracy of 0.837. Although scaling had little impact on the tree-based models, StandardScaler was applied to all algorithms to maintain a consistent preprocessing approach and enhance KNN's performance.

TABLE V. SCALING METHOD COMPARISON

Algorithm	Accuracy	Precision	Recall	F1-score	AUC-ROC
DT					
StandardScaler	0.905303	0.846154	0.835443	0.840764	0.889497
MinMaxScaler	0.905303	0.846154	0.835443	0.840764	0.889497
MaxAbsScaler	0.905303	0.846154	0.835443	0.840764	0.889497
KNN					
StandardScaler	0.837121	0.772727	0.645570	0.703448	0.879405
MinMaxScaler	0.787879	0.676923	0.556962	0.611111	0.827095
MaxAbsScaler	0.814394	0.734375	0.594937	0.657343	0.858194
RF					
StandardScaler	***0.935606***	***0.955882***	***0.822785***	***0.884354***	***0.981594***
MinMaxScaler	***0.935606***	***0.955882***	***0.822785***	***0.884354***	***0.981594***
MaxAbsScaler	***0.935606***	***0.955882***	***0.822785***	***0.884354***	***0.982005***

D. Results after Applying Feature Selection Using Cox

The DT classifier, using feature selection with the Cox model, achieved strong performance with high accuracy, AUC-ROC, and recall, indicating effective identification of death events. The balanced precision and F1-score reflect reliable performance, though nine false negatives suggesting the need for data balancing to further improve recall and reduce missed critical events. Choosing a refined feature set strengthened the model's ability to cope with class imbalance and raised its predictive accuracy. Fig. 7 shows the decision-tree classifier's confusion matrix after features were selected using the Cox proportional hazards method.

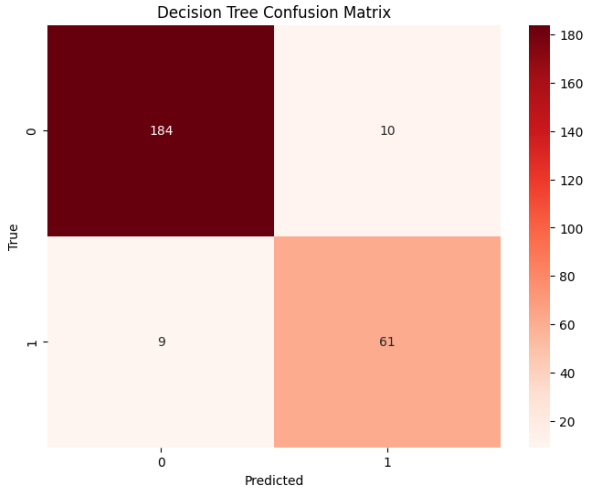

Fig. 7. Confusion matrix for the DT classifier after feature selection using the CoxPH model.

Fig. 8 represents the AUC-ROC curve for the DT classifier after feature selection using the CoxPH model.

The KNN classifier, using feature selection with the Cox model, demonstrated moderate predictive performance. While the model shows reasonable capability in distinguishing between classes, its limited recall highlights challenges with imbalanced data. The trade-off between FP and FN underscores the need for improved sensitivity to the minority class. These findings suggest that data balancing techniques could further enhance the model's ability to capture critical events effectively. Fig. 9 represents the confusion matrix for KNN classifier after feature selection using the CoxPH model.

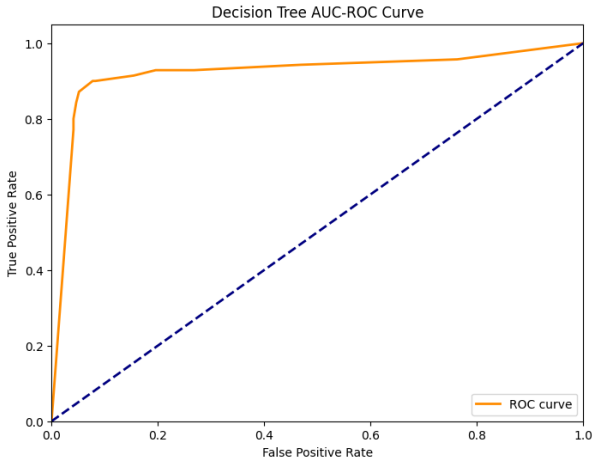

Fig. 8. AUC-ROC curve for the DT classifier after feature selection using the CoxPH model.

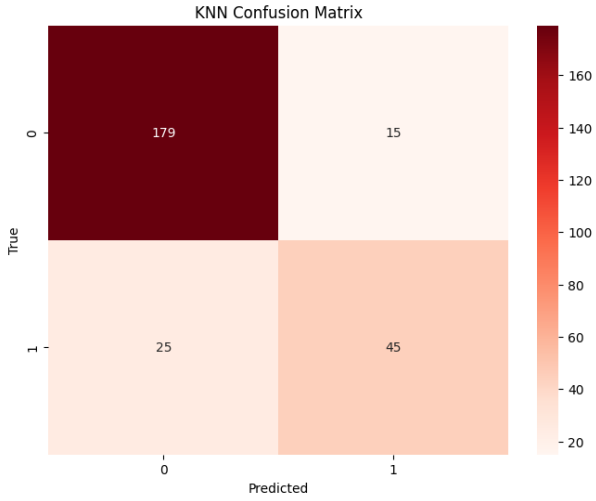

Fig. 9. Confusion matrix for the KNN classifier after feature selection using the CoxPH model.

Fig. 10 represents the AUC-ROC curve for the KNN classifier after feature selection using the CoxPH model.

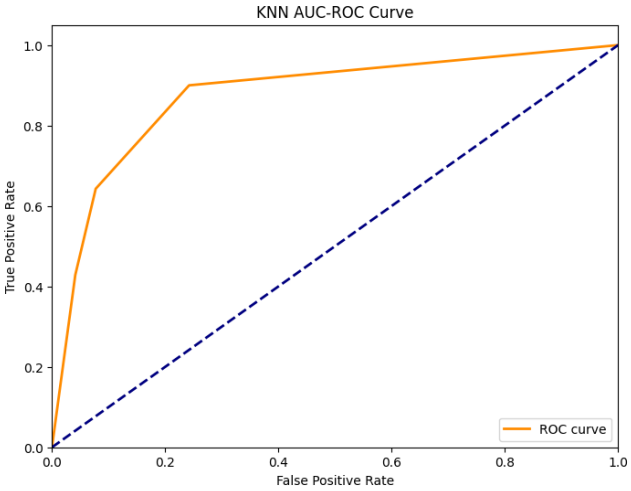

Fig. 10. AUC-ROC curve for the KNN classifier after feature selection using the CoxPH model.

The RF classifier, using feature selection with the Cox model, demonstrated moderate predictive performance. With highly accurate, near-perfect AUC-ROC, and precisely balanced precision and recall, the model is working effectively in both correctly predicting death occurrences and non-occurrences and in having low FP and FN rates. These findings are a proof of the robustness of Random Forest in the context of handling imbalanced data, yet data balancing would enhance sensitivity to critical events even more. Model reliability and performance were greatly enhanced by feature selection. Fig. 11 represents the confusion matrix for RF classifier after feature selection using the CoxPH model.

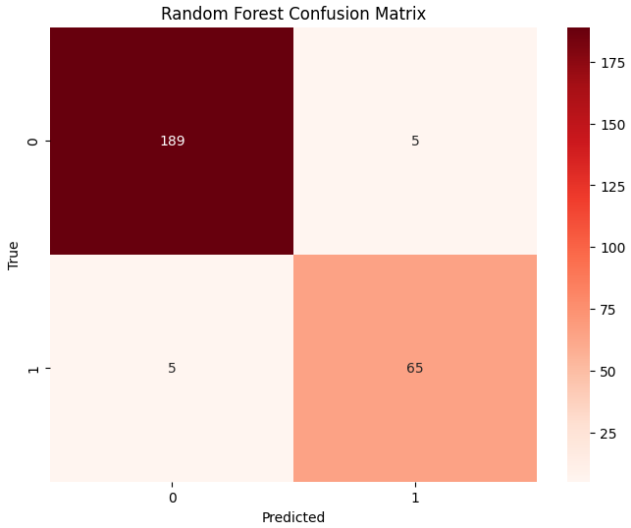

Fig. 11. Confusion matrix for the RF classifier after feature selection using the CoxPH model.

Fig. 12 represents the AUC-ROC curve for the RF classifier after feature selection using the CoxPH model.

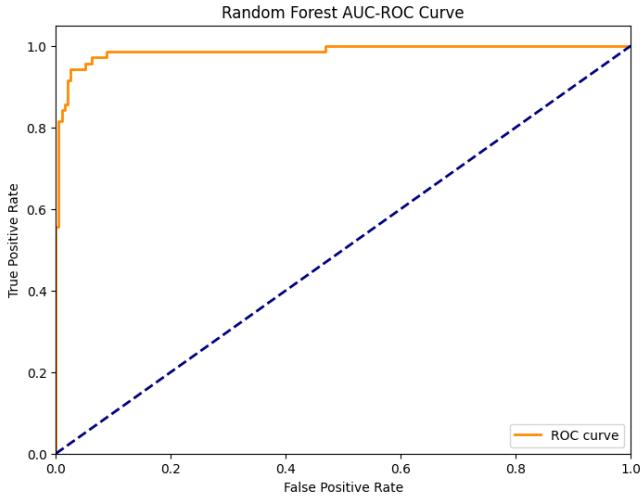

Fig. 12. AUC-ROC curve for the RF classifier after feature selection using the CoxPH model.

CoxPH feature selection greatly improved the classifier's performance by focusing on the leading features like age, anemia, and creatinine. Random Forest was also the best-performing model at the same time.

V. Discussion

The research examined the performance of three classifiers (DT, KNN, and RF) as shown in Table VI. It employed feature selection using the Cox proportional hazards model and various data balancing methods (Random Over-Sampling, SMOTE, and Random Under-Sampling). The findings show that the application of feature selection achieves better predictions in healthcare datasets.

The comparative analysis of three ML classifiers (DT, KNN, and RF) - for heart failure survival prediction revealed significant insights into model performance and feature selection effectiveness as presented in Table VII. The Random Forest classifier demonstrated superior predictive capability across all evaluation metrics when using the original unbalanced dataset, achieving 96.2% accuracy, 92.9% F1-score, and 0.987 AUC-ROC values. This exceptional performance shows that the ensemble nature of RF, with its inherent feature randomness and bootstrap aggregation, provides robust predictive power even without explicit class balancing techniques. The application of the CoxPH model for feature selection proved particularly valuable, as it enhanced Random Forest's performance by identifying and retaining only the most clinically relevant predictors. The selected features - including “age”, “ejection fraction”, “serum creatinine”, and “high blood pressure” - represent well-established risk factors in cardiovascular medicine, which likely contributed to the model's strong discriminative ability. This feature selection process not only improved model accuracy but also increased clinical interpretability by focusing on medically meaningful variables. While data balancing methods like SMOTE and random under-sampling showed some capacity to improve recall metrics, they generally came at the cost of reduced precision in the RF model. The minimal performance improvement from balancing suggests that Random Forest's

inherent mechanisms for handling class imbalance may be sufficient for this particular dataset. The Decision Tree classifier showed respectable performance but consistently underperformed compared to Random Forest, likely due to its simpler structure and greater susceptibility to overfitting. The KNN algorithm demonstrated the weakest performance among the three classifiers, a finding that aligns with expectations given its known sensitivity to high-dimensional data and class imbalance. The superior performance of tree-based methods in this medical prediction task reinforces their established utility in healthcare analytics, where they often provide an effective balance between predictive accuracy and model interpretability. The study titled "Prediction of Survival of Heart Failure Patients Using Random Forest" by Sri Rahayu et al. evaluates RF, DT, KNN, SVM, ANN, and Naïve Bayes using resample and SMOTE techniques, achieving its best accuracy of 94.31% with RF and resampling [7]. However, it lacks explicit feature selection and detailed metrics evaluation beyond accuracy. The study titled "Improving the Prediction of Heart Failure Patients' Survival Using SMOTE and Effective Data Mining Techniques" by Abid Ishaq et al. employs nine classifiers, including DT, AdaBoost, RF, and ETC, with SMOTE for data balancing and Random Forest for feature selection [6]. It achieves its best accuracy of 92.62% with ETC but does not explore multiple balancing techniques. In contrast, our study stands out by combining the CoxPH Model for feature selection with a comprehensive evaluation of three ML models. RF model achieves a higher performance than other studies.

TABLE VI. SUMMARY OF PERFORMANCE METRICS FOR THE DT, KNN AND RF CLASSIFIERS USING FEATURE SELECTION WITH THE COXPH MODEL AND VARIOUS DATA BALANCING TECHNIQUES

Classifier	Balancing Method	Accuracy	Precision	Recall	F1-score	AUC-ROC
Decision Tree	***Without sampling***	0.93	0.86	0.87	0.87	0.92
	Random Over-Sampling	0.90	0.78	0.91	0.84	0.90
	SMOTE	0.90	0.84	0.85	0.84	0.91
	Random Under-Sampling	0.89	0.78	0.87	0.83	0.90
KNN	***Without sampling***	0.85	0.75	0.64	0.69	0.87
	Random Over-Sampling	0.84	0.70	0.80	0.75	0.88
	SMOTE	0.83	0.70	0.80	0.74	0.87
	Random Under-Sampling	0.80	0.63	0.84	0.72	0.89
Random Forest	***Without sampling***	0.962	0.93	0.93	0.93	0.987
	Random Over-Sampling	0.939	0.90	0.89	0.90	0.98
	SMOTE	0.943	0.93	0.87	0.90	0.98
	Random Under-Sampling	0.92	0.82	0.92	0.87	0.975

TABLE VII. COMPARATIVE PERFORMANCE ANALYSIS OF HEART FAILURE PREDICTION MODELS

Study and Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC	Key Methodology
Rahayu et al. (2020) [7]						Resample + SMOTE and No feature selection
Random Forest (RF)	94.31%	-	0.943	-	0.976	
Decision Tree (DT)	87.29%	-	0.873	-	0.872	
KNN	86.95%	-	0.870	-	0.816	
Ishaq et al. [6]						SMOTE + RF Feature Selection
Extra Trees (ETC)	92.62%	0.93	0.93	0.93	-	
Random Forest (RF)	91.80%	0.92	0.92	0.92	-	
Our Study						CoxPH
Random Forest (RF)	**96.2%**	0.93	0.93	0.93	**0.987**	Cox feature selection

The comparative analysis highlights the performance of various heart failure prediction models across multiple studies. Rahayu et al. achieved their highest accuracy of 94.31% using a Random Forest model with SMOTE and resampling techniques, though their approach did not involve explicit feature selection [7]. Ishaq et al. implemented both SMOTE and RF-based feature selection, with the Extra Trees Classifier achieving 92.62% accuracy and balanced precision, recall, and F1-score values of 0.93 [6]. In contrast, our study outperformed prior work by leveraging the CoxPH model for feature selection, enabling the RF classifier to achieve 96.2% accuracy, which is higher than that of Rahayu et al. by more than 1.5% and that of Ishaq et al. by more than 3% and the highest AUC-ROC value of 0.987. These results underscore the value of integrating clinically meaningful feature selection with robust ensemble models to enhance predictive performance in heart failure prognosis.

It is also noteworthy that prior works did not incorporate survival analysis in their classification frameworks (with the exception of Lee et al., who focused on Cox regression alone). Our approach demonstrates that using survival analysis not only contributes to understanding which features are important over time but also improves the selection of features for classification models, thus carrying the strengths of both statistical survival methods and machine learning.

VI. Conclusion

This study showed the effectiveness of integrating survival analysis and machine-learning methods in predicting survival outcomes for heart-failure patients. By applying the CoxPH model, we identified the most important clinical features and used them to train and evaluate DT, KNN and RF classifiers. Among these, the RF model outperformed the others, achieving notable accuracy and discriminative power (AUC-ROC = 0.987). The combination of clinically relevant feature selection, careful preprocessing and systematic hyper-parameter tuning produced models that balance accuracy with interpretability, underscoring the promise of hybrid predictive frameworks for early diagnosis and data-driven decision-making in heart-failure care.

Despite the strong performance, this study has limitations. It is based on a single dataset, which may limit generalizability to other populations or clinical settings. The current models do not incorporate temporal or longitudinal patient data beyond the static features available in the dataset. Additionally, model interpretability—while improved through feature selection—still lacks integration with clinician-friendly interfaces or visualization tools. Addressing these limitations in future studies (such as validating on external cohorts, including time-series data, and developing clinician-facing explainable AI dashboards) will help strengthen the applicability and trust in such predictive tools.

While this study presents promising results, it is important to note that some aspects, such as the use of data balancing techniques and the scope of model evaluation, could be further enhanced in future work. Expanding the dataset and exploring more advanced architectures may lead to even better performance and broader applicability. Future studies are encouraged to test deep-learning architectures, replicate findings on external cohorts for greater generalizability, and integrate explainable-AI dashboards that clinicians can use at the point of care to visualise individual risk trajectories in real time.

References

[1] A. K. Malakar, D. Choudhury, B. Halder, P. Paul, A. Uddin and S. Chakraborty, "A review on coronary artery disease, its risk factors, and therapeutics," Journal of cellular physiology, vol. 234, no. 10, p. 16812–16823, 2019.

[2] H. Benhar, A. Idri and J. Fernández-Alemán, "Data preprocessing for heart disease classification: A systematic literature review," Computer methods and programs in biomedicine, vol. 195, pp. 105-123, 2020.

[3] M. Benllarch, S. El Hadaj and M. Benhaddi, "Improve Extremely Fast Decision Tree Performance through Training Dataset Size for Early Prediction of Heart Diseases," in 4th International Conference on Systems of Collaboration Big Data, Internet of Things & Security, 2019.

[4] E. Owusu, P. Boakye-Sekyerehene, J. Appati and J. Y. Ludu, "Computer-Aided Diagnostics of Heart Disease Risk Prediction Using Boosting Support Vector Machine," Computational intelligence and neuroscience, pp. 1-12, 2021.

[5] A. Kilic, "Artificial Intelligence and Machine Learning in Cardiovascular Health Care," The Annals of thoracic surgery, vol. 109, no. 5, p. 1323–1329, 2020.

[6] A. Ishaq, M. Umer, S. Sadiq, S. Mirjalili, V. Rupapara, S. Ullah and M. Nappi, "Improving the Prediction of Heart Failure Patients Survival Using SMOTE and Effective Data Mining Techniques," IEEE Access, vol. 9, pp. 39707-39716, 2021.

[7] S. Rahayu, J. J. Purnama, A. B. Pohan, F. S. Nugraha, S. Nurdiani and S. Hadianti, "PREDICTION OF SURVIVAL OF HEART FAILURE PATIENTS USING RANDOM FOREST," Pilar Nusa Mandiri, vol. 16, no. 2, pp. 255-260, 2020.

[8] O. O. Oladimeji and O. Oladimeji, "Predicting Survival of Heart Failure Patients Using Classification Algorithms," Journal of Information Technology and Computer Engineering (JITCE), vol. 4, no. 2, pp. 90-94 , 2020.

[9] H. J. Lee, S.-S. Yoo and K.-Y. Lee, "Machine Learning-Enhanced Survival Analysis: Identifying Significant Predictors of Mortality in Heart Failure," KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS, vol. 18, no. 9, pp. 2495-2511, 2024.

[10] M. Mamun, A. Farjana, M. Al Mamun, M. M. Rahman and M. S. Ahammed, "Heart failure survival prediction using machine learning algorithm: am I safe from heart failure?," 2022 IEEE World AI IoT Congress (AIIoT), pp. 194-200, 2022.

[11] A. Spooner, E. Chen, A. Sowmya, P. Sachdev, N. A. Kochan, J. Trollor and H. Brodaty, "A comparison of machine learning methods for survival analysis of high-dimensional clinical data for dementia prediction.," Scientific Reports, vol. 10, 2020.

[12] G. Menardi and N. Torelli, "Training and assessing classification rules with imbalanced data," Data Mining and Knowledge Discovery, vol. 28, p. 92–122, 2014.

[13] R. Blagus and L. Lusa, "Joint use of over- and under-sampling techniques and cross-validation for the development and assessment of prediction models," BMC Bioinformatics, vol. 16, no. 1, p. 1–10, 2015.

[14] N. Lunardon, G. Menardi and N. Torelli, "ROSE: A Package for Binary Imbalanced Learning," R Journal, vol. 6, 2014.

[15] I. D. Mienye and N. Jere, "A Survey of Decision Trees: Concepts, Algorithms, and Applications," IEEE Access, vol. 12, pp. 86716 - 86727, 2024.

[16] A. Curth, A. Jeffares and M. van der Schaar, "Why do Random Forests Work? Understanding Tree Ensembles as Self-Regularizing Adaptive Smoothers," arXiv, 2024.

[17] P. Cunningham and S. J. Delany, "k-Nearest Neighbour Classifiers: 2nd Edition (with Python examples)," arXiv, 2020.

[18] S. Swaminathan and B. R. Tantri, "Confusion Matrix-Based Performance Evaluation Metrics," African Journal of Biomedical Research, vol. 27, pp. 4023-4031, 2024.

[19] A. Velu and A. Alexia, "Heart Failure Prediction - Clinical Records," Kaggle, 2024. [Online]. Available: https://www.kaggle.com/datasets/aadarshvelu/heart-failure-prediction-clinical-records. [Accessed 2025].

[20] A. Asuncion and D. J. Newman, "UCI machine learning repository," University of California, California, Irvine, 2007.

[21] T. Ahmad, A. Munir, S. H. Bhatti, M. Aftab and M. A. Raza, "Survival analysis of heart failure patients: A case study," PLoS ONE, vol. 12, no. 7, 2017.