Advances in Basic and Applied Sciences 5 (2025) 29-37

Advances in Basic and Applied Sciences

journal homepage: https://abas.journals.ekb.eg/

Effective heart disease diagnosis accuracy through hybrid machine learning methods

A.M.M. Madbouly *

Mathematics Department, Faculty of Science, Helwan University, Helwan, Egypt

ARTICLE INFO

Article history:
Received 14 August 2025
Received in revised form 25 August 2025
Accepted 26 August 2025
Available online 1 September 2025

10.21608/ABAS.2025.411130.1074

Keywords: Heart diseases , machine learning, artificial intelligence.

ABSTRACT

For many years, heart disease has been the leading cause of death worldwide. This highlights the urgent need for reliable, practical methods for early detection of heart disease for early treatment. In the healthcare system, data mining has become a widely used tool for handling massive amounts of data. Researchers are using various data mining and machine learning techniques to analyze complex medical datasets, helping healthcare professionals address heart disease earlier. This study uses different supervised learning to build models for heart conditions. The analysis makes use of a dataset called Cleveland from UCI Machine Learning Repository, which has 303 entries and 76 characteristics. However, only 14 critical attributes are chosen for model evaluation to ensure meaningful performance comparisons. The prime goal of this research is to estimate the likelihood of heart disease in patients. Hybrid techniques such as naïve bayes, support vector machine, and knn can improve prediction performance beyond traditional algorithms.

1. Introduction

Cardiovascular disease has been a leading cause of death worldwide over the past decade. According to the World Health Organization, approximately 19.8 million people die each year from cardiovascular disease. [1]. Personal and professional habits are to be blamed for heart disease as well, with a genetic predisposition. Common lifestyle-related risk factors encompass smoking, heavy intake of caffeine and alcohol, chronic stress, and lack of physical exercise. Additionally, physiological conditions such as obesity, hypertension, elevated cholesterol levels, and existing cardiovascular disease can further contribute to health risks. Early and accurate diagnosis of complications in health related to the heart are important not only for better prevention of

* **Corresponding author E-mail:** @science.helwan.edu.eg

death, but also for being able to treat patients properly [2].

Data mining is widely applied in diverse fields, including healthcare, commerce, and education. Machine learning is recognized as one of the most rapidly advancing domains within artificial intelligence. Extensive datasets originating from various sectors, such as health sciences, can be examined through these algorithms. This approach serves as an alternative to conventional predictive modeling methods, wherein computational systems are employed to identify complex and non-linear relationships among multiple variables, thereby minimizing discrepancies between projected and actual results. [3]. Data mining involves analyzing extensive datasets to uncover valuable insights that support informed decision-making, particularly from historical data for future applications. In healthcare, the abundance of patient data necessitates the use of diverse machine learning techniques to effectively extract meaningful patterns. These computational methods enable healthcare practitioners to improve diagnostic accuracy and clinical outcomes. Specifically, employing classification algorithms in medical data mining aids in the prediction and diagnosis of heart disease, thereby providing crucial support for clinical decision-making processes. [4].

Data mining refers to the process of deriving meaningful information from extensive datasets. In the context of heart disease prediction, various techniques are applied. A comparative evaluation of these classification approaches is introduced as mentioned in [5]. UCI source [6] was used in this investigation. Model of classification is created using different classification techniques to predict cardiac disease. This study provides an overview of algorithms used for cardiac disease prediction, as well as a comparison of existing systems. The article also discusses future research and development potential. In addition, this study will seek to establish a hybrid strategy by combining multiple categorization algorithms in order to improve forecast accuracy. Previous study has demonstrated that combining classifiers like as decision trees, random forest, and logistic regression can improve the accuracy of heart disease prediction. Future research will concentrate on optimizing these combinations in order to maximize the strengths of each method and potentially produce more accurate results in practical applications.

In this study, multiple machine learning approaches are used to predict cardiac disease. To provide background, machine learning and commonly used categorization techniques are briefly described below.

Machine learning is a growing branch of artificial intelligence. Its fundamental goal is to develop systems that can learn, and forecast based on their experiences. It creates models by training machine learning algorithms on a training dataset. The model uses the new data to forecast heart disease. It uses machine learning to find hidden patterns in the input dataset and then builds models. It generates accurate forecasts for new datasets. The dataset has been cleaned and missing values filled. The model predicts heart disease using the new input data and then tests its accuracy.

Machine learning methods are categorized as:

Supervised Learning

The model is trained using a labelled dataset. It has data and outputs. Data is classified and separated into training and testing datasets. Our model is trained using the training dataset, while the testing dataset is used to generate new data in order to improve model accuracy. The dataset includes models and their output. It uses categorization and regression as examples [7].

Unsupervised Learning

Unsupervised learning algorithms operate on unlabeled datasets, where no ground truth classifications or target variables are provided during the training process. The primary objective of these methods is to discover latent structures and patterns within the data distribution. The algorithmic framework learns to extract meaningful representations that can reveal underlying data characteristics without explicit supervision. When presented with novel input instances, the trained model applies learned pattern recognition capabilities to identify similar structural relationships, subsequently utilizing these discovered patterns to generate insights about the data's inherent organization. This methodology does not produce direct output predictions or responses, as the focus remains on exploration data analysis and structure identification. Clustering algorithms exemplify this unsupervised learning paradigm by partitioning data points into distinct groups based on similar measures and distance metrics. [8].

Reinforcement Learning method.

Reinforcement learning constitutes a computational learning paradigm wherein an autonomous agent acquires optimal behavioral policies through dynamic environmental interaction, eschewing the requirement for pre-annotated training datasets. This methodology diverges fundamentally from supervised learning approaches by implementing an exploratory learning framework where the agent iteratively refines its decision-making processes through experiential feedback mechanisms [9]. The learning system receives

environmental responses in the form of scalar reward signals or penalty functions, which serve as optimization objectives for policy gradient updates. Through sequential action selection and consequence evaluation, the agent employs temporal difference learning to maximize cumulative expected returns across extended time horizons. This approach proves particularly efficacious in domains characterized by sparse or unavailable labeled training data, yet where performance metrics can be quantitatively defined through reward structures.

Conversely, classification methodologies represent core supervised learning techniques that leverage annotated training corpora to construct predictive models capable of estimating posterior probabilities for discrete outcome categories, such as cardiovascular pathology diagnosis. Algorithmic approaches including decision tree induction, Naïve Bayes probabilistic classifiers, and support vector machine optimization utilize patient feature vectors in conjunction with established diagnostic labels to generate predictive models. These trained classifiers subsequently enable probabilistic risk assessment for cardiovascular disease manifestation in previously unseen patient populations through learned feature-outcome associations [10].

2. Classification Machine Learning Techniques

The categorization task is employed to estimate subsequent cases based on recent data.[11]. Cardiovascular disease diagnostic systems have been extensively investigated through the implementation of various data mining algorithms, including Naïve Bayes probabilistic classifiers, artificial neural networks, and decision tree-based learning models. These computational approaches have demonstrated efficacy in automated cardiac pathology identification through pattern recognition and statistical inference methods. The predictive performance metrics of these algorithmic frameworks exhibit sensitivity to feature dimensionality, with classification accuracy correlating with the cardinality of input attributes utilized in the model training process. Empirical evaluations reveal that diagnostic precision varies as a function of the selected feature subset, indicating the critical importance of feature selection and dimensionality optimization in cardiovascular risk assessment applications.

3. Literal review

According to [12], linear classifier can work accurately with Heart risk factors dataset. In this section we will show the common classifier that works on this data as follows:

3.1. Naïve Bayes

The Naïve Bayes method is a supervised learning algorithm that applies Bayes' theorem for categorization tasks. It operates under the hypothesis of strong (naive) independence between features. This means it treats each attribute as unrelated to the others, with no correlation, and considers their contributions to the final prediction independently to maximize the overall probability. While it is called a Bayesian model, it does not implement full Bayesian methods [13].

The Naïve Bayes classifier implements Bayesian probabilistic inference to compute posterior class probabilities given a vector of predictor variables, utilizing Bayes' theorem as its foundational mathematical framework. This probabilistic algorithm exhibits widespread adoption across diverse application domains owing to its computational simplicity, straightforward implementation requirements, and computational efficiency when processing both linear and non-linear datasets of considerable complexity. Nevertheless, the classifier's predictive performance may be constrained by its fundamental assumption of conditional independence among feature variables, which rarely holds in real-world data distributions. Empirical evaluation using Support Vector Machine Recursive Feature Elimination (SVM-RFE) for optimal feature subset selection yielded a classification accuracy of 84.1584% when employing the ten most discriminative predictor variables, demonstrating the algorithm's practical utility in supervised learning tasks despite its theoretical limitations. [14]. Alternatively, when utilizing the complete feature set comprising all thirteen attributes from the Cleveland cardiovascular disease dataset, the Naïve Bayes classifier achieved a classification accuracy of 83.49%, indicating a marginal decrease in predictive performance compared to the optimized feature subset approach. [15]. We achieved an accuracy of 87% when applied to the full set of thirteen features in the Cleveland dataset.

3.2. Decision Tree

Decision tree algorithms constitute a versatile classification methodology capable of processing both categorical and continuous numerical variables within a unified framework. The algorithm constructs a hierarchical tree-like data structure through recursive binary partitioning, rendering it computationally intuitive and broadly applicable across diverse analytical domains, with particular efficacy demonstrated in medical data analysis applications. This algorithmic approach offers straightforward implementation procedures while

providing transparent, interpretable model representations through graphical tree visualization, facilitating comprehension of the underlying decision-making logic and feature importance hierarchies within the classification process. The structure of a decision tree consists of three types of nodes:

- Root node: The primary node from which the tree originates, and all decisions branch out.
- Internal (interior) nodes: These nodes evaluate different attributes within the dataset.
- Leaf nodes: These represent the final outcome or classification after testing conditions.

The procedure splits data into two or more similar subsets using the most significant features. It calculates the entropy of each attribute and selects those with the highest information gain (or lowest entropy) to perform the split, allowing for more accurate classification.

$$\text{Entropy}(S) = \sum_{i=1}^{c} P_i \log_2 P_i$$

$$\text{Gain}\,(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v)$$

As a researcher, the findings indicate that the outputs produced by this algorithm are straightforward to interpret, making them user-friendly for analysis [16]. The decision tree method typically achieves superior accuracy compared to many other algorithms, owing to its ability to represent the dataset in a tree-structured model. Nevertheless, decision trees can sometimes lead to over-classification, and at each decision point, only one attribute is evaluated. For instance, Chauhan et al. [17] reported an accuracy of 71.43% using the decision tree algorithm, while another study demonstrated considerably lower performance, achieving just 42.90% accuracy [18].

3.3. Random Forest Algorithm

Random forest algorithms represent an ensemble-based supervised learning methodology that constructs multiple decision tree classifiers to form a collective predictive model. This bagging approach generates numerous independent decision trees, each contributing a classification vote, with the final prediction determined through majority voting consensus across the entire ensemble. The predictive accuracy of random forest models typically exhibits positive correlation with forest size, as increased tree population enhances model robustness and generalization capability.

The algorithm employs three primary sampling strategies: Random Input Selection (Forest RI), which randomly selects feature subsets for each tree; Random Combination (Forest RC), which creates random linear combinations of features; and hybrid approaches that integrate both methodologies. While applicable to both classification and regression tasks, random forest demonstrates superior performance in categorical prediction problems and exhibits inherent resilience to missing data through its bootstrap aggregating mechanism.

However, the computational complexity of random forest algorithms scales significantly with dataset size and ensemble cardinality, resulting in increased training time and prediction latency. This computational overhead, coupled with the ensemble's inherent complexity, reduces model interpretability compared to single decision tree approaches, creating trade-offs between predictive performance and algorithmic transparency in machine learning applications. For example, the random forest accomplished an accuracy of 91.6% on the Cleveland heart disease dataset [19], while it achieved 97% accuracy on a dataset referred to as the People's dataset [12].

3.4. K-Nearest Neighbor (K-NN)

The k-nearest neighbors algorithm represents a lazy learning paradigm within supervised machine learning that performs classification through local neighborhood analysis. This non-parametric approach assigns class labels to test instances by examining the class distribution among the most similar training examples within the feature space. As a memory-based learning technique, KNN defers computational processing until query time, making classification decisions based solely on the local structure of the training data surrounding each test point. Distance-based similarity assessment typically utilizes Euclidean metrics to establish proximity relationships between feature vectors, enabling the algorithm to identify the nearest training instances and subsequently apply majority voting principles to determine the predicted class label for unlabeled observations [3].

To categorize a new data point, the K-NN algorithm looks at a set of labelled data points. It groups data based on similarity, making it successful in imputing missing values. After filling in the missing data, multiple predicting methods can be used to the dataset, and combining different algorithms can enhance accuracy. K-NN is simple to implement because it does not need the creation of a model or the formulation of data assumptions. Its versatility makes it appropriate for classification, regression, and search problems. Despite its simplicity, the introduction of noisy or irrelevant information can impair its accuracy. Pouriyeh et al. found

an accuracy of 83.16% when using K = 9 [15].

3.5. Logistic Regression

Logistic regression is a widely utilized statistical technique for classification problems, which estimates the probability of a categorical outcome based on one or more independent variables. This model employs the logistic, or sigmoid, function to transform a linear combination of predictor variables into a probability value ranging between zero and one [20]. Logistic regression is usually used in binary and multi-class classification topics across a varied area range, including healthcare, finance, and social sciences, due to its simplicity, interpretability, and efficiency. Despite its efficiency, logistic regression implies a linear relationship between predictors and log-odds of the outcome, which limits its ability to detect complicated non-linear patterns in data.

3.6. Support Vector Machine (SVM)

Support Vector Machine (SVM) is an advanced supervised learning algorithm designed to identify the optimal hyperplane that distinctly divides different classes within a high-dimensional feature space by maximizing the margin between them[21]. Support Vector Machines (SVM) are highly effective for classification tasks involving clearly defined and separable categories, as they concentrate on the critical data points nearest to the decision boundary, referred to as support vectors. The technique's use of diverse kernel functions allows it to capture both linear and complex non-linear relationships within the data. SVM has demonstrated strong performance across numerous domains such as image recognition, bioinformatics, and text classification, exhibiting robustness against overfitting, particularly in environments with high-dimensional feature spaces [11].

3.7. Gradient Boosting

Gradient Boosting is an ensemble learning technique that constructs additive predictive models by iteratively training weak learners, typically decision trees, each one aimed at correcting the errors made by the preceding models. [22]. This iterative technique improves prediction accuracy by merging numerous weak predictors. Gradient boosting algorithms are recognized for their ability to handle a wide range of data and complexities, making them useful for classification and regression applications. However, to avoid overfitting and obtain optimal performance, hyperparameters like learning rate, tree depth, and number of estimators must be precisely tuned.

3.8. MLP Neural Network (Multilayer Perceptron)

The Multilayer Perceptron (MLP) is a type of feedforward artificial neural network consisting of many layers of consistent neurons that grow hierarchical representations of incoming data [23]. Each neuron has a non-linear activation function, which allows the network to simulate complicated, non-linear interactions between characteristics and targets. MLPs are widely utilized for classification and regression issues, serving as the foundation for deep learning architecture. Their adaptability and ability to learn complex patterns make them ideal for applications such as speech recognition, natural language processing, and medical diagnosis [11].

The perceptron algorithm learns from a set of training examples by repeatedly processing the dataset until it finds a weight vector that properly classifies all of the training cases. Once the optimal prediction algorithm has been identified, it is used to predict the labels of the test data [11].

3.9. XGBoost

XGBoost (Extreme Gradient Boosting) is a highly efficient and scalable implementation of gradient boosting that incorporates regularization methods and leverages parallel computing to enhance both training efficiency and predictive performance [24]. Extreme Gradient Boosting represents an optimized ensemble learning framework that builds upon traditional gradient boosting methodologies through the integration of advanced regularization techniques and distributed computational architectures. This machine learning algorithm enhances model training velocity and classification accuracy by implementing parallel processing capabilities alongside sophisticated overfitting prevention mechanisms, resulting in improved generalization performance across diverse datasets [25].

3.10. LightGBM

LightGBM is a gradient boosting framework constructed with productivity and scalability in mind, using a histogram-based algorithm and a leaf-wise tree growth technique [26]. Compared to classic boosting approaches, these advances allow for faster training and lower memory consumption. LightGBM excels at large-scale datasets with high-dimensional features and complicated interactions, frequently surpassing other boosting methods. Its ability to handle categorical features natively and distribute training across numerous machines adds to its appropriateness for production scenarios and big data applications [27].

4. Proposed method

The goal of this project is to use computerized diagnostic procedures to forecast the possibility of cardiac

disease, which will benefit both clinicians and patients. To achieve this goal, we use a range of machine learning techniques on a heart disease dataset and offer a detailed analysis of the results. In addition, we investigate merging multiple classifiers to improve prediction accuracy and overall model performance**.**

This approach leverages the strengths of multiple models (Naive Bayes, SVM, and KNN) by combining them with weighted soft voting (1,2,1). By assigning more weight to the better-performing SVM, the ensemble aims to improve overall predictive accuracy on the test dataset, potentially outperforming single models. This technique is a common way to boost classification performance via model diversity. We note also if we increase weights like (1,5,1) accuracy will decrease from 88% to 86%. It means keep SVM weighing twice as much of naïve Bayes and kNN.

We will illustrate what the following algorithm did us.

1. start
2. **Assign Column Names**

   Define and assign meaningful names to each column for better dataset interpretation.
3. **Load the Dataset**

   Import the heart disease dataset from a CSV file.
4. **Identify and Replace Missing Values**

   Replace placeholders (e.g., '?') used for missing values with proper null indicators (e.g., Nan).
5. **Convert Data Types**

   Ensure all columns are converted to numeric types to support machine learning algorithms.
6. **Handle Missing Values**

   Address missing data by either:
   - Dropping rows with missing values (used in this workflow), or
   - Imputing missing values (e.g., using median).
7. **Recode Target Variable**

   Convert the multi-class target variable into a binary classification format:

o 0 indicates no heart problems.
o 1 indicates the presence of cardiac disease.

8. **Define train and test data**
9. **Apply a machine learning technique**
10. **Compute accuracy**
11. **End**

5. Dataset

In this study, datasets from the UCI Machine Learning Repository were employed. The dataset is composed of 300 authentic instances described by 14 attributes—13 predictors and one target variable—including age, maximum heart rate, gender, cholesterol level, chest pain type, Thalassemia test result, blood pressure, ST depression, and other relevant factors.

In the subsequent sections, an examination of the experimental data is presented. As previously stated, the analysis was conducted using the Cleveland dataset[12]. This experiment includes different eleven machine learning techniques. 80% of the dataset size is used as training, while the remaining (20%) is used as test. Experimental results show the accuracy in Table 1.

Table 1: Machine learning model and their accuracies

Machine learning algorithm	Accuracy
KNN	85%
SVM	87%
Naïve bayes	87%
Logistic Regression	82%
MLP Neural Net	82%
LightGBM	82%
Random forest	80%
XGBoost	80%
Decision Tree	77%
Gradient Boosting	77%
Proposed method	88%

6. Results and Analysis.

The main objective of the research is early detection of heart diseases using machine learning, due to its importance in the treatment journey. This study focused on supervised machine learning classification methods utilizing algorithms such as K-Nearest Neighbors (KNN), Neural Networks, Support Vector Machines

(SVM), Naive Bayes, Logistic Regression, Multilayer Perceptron (MLP) Neural Networks, LightGBM, Random Forest, XGBoost, Decision Trees, and Gradient Boosting, applied to the UCI dataset repository. Multiple experiments were conducted using Jupyter Notebook, where the dataset was preprocessed and split into training (80%) and testing (20%) subsets. Various classification techniques were then employed to assess prediction accuracy.

The comparative results, illustrated in Figure 1, reveal distinct differences in the performance of these algorithms. The proposed approach achieved the highest accuracy of 88%, indicating its superior effectiveness for this dataset. Both the Support Vector Machine and Naive Bayes classifiers followed closely, each attaining an accuracy of 87%, reflecting their well-known strength and reliability in classification tasks.

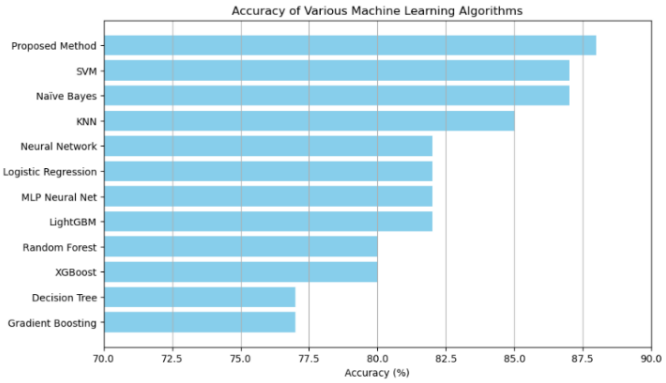

Fig. 1: Accuracy of various machine learning models.

K-Nearest Neighbors demonstrated an accuracy of 85%, highlighting its efficacy in leveraging feature-space proximity for classification. Meanwhile, Neural Networks, Logistic Regression, MLP Neural Networks, and LightGBM all reached an accuracy level of 82%, suggesting that traditional statistical models and certain advanced gradient boosting methods offer comparable predictive capabilities in this context. Random Forest and XGBoost both show moderate classification performance, each achieving an accuracy of 80%. Although these results are slightly lower than the highest-performing models, they still reflect the well-established strengths of these algorithms, particularly in capturing complex non-linear patterns and interactions among features. Conversely, Decision Tree and Gradient Boosting methods recorded the lowest accuracy rates at 77%, which could be attributed to potential overfitting or suboptimal parameter tuning in this particular study.

Overall, the data suggest that ensemble or hybrid approaches, which leverage the complementary advantages of multiple algorithms—as evidenced by the proposed method—may provide improved predictive accuracy and robustness. It also emphasizes the importance of selecting an algorithm aligned with data characteristics and employing robust tuning and feature engineering for optimal results.

This comparative analysis aligns with literature findings where SVM, Naïve Bayes, and KNN often rank highly in classification accuracy, while decision trees and simpler boosting methods may underperform without careful optimization. Future work could explore further enhancements through ensemble methods and deep learning architectures to push accuracy boundaries.

Conclusion

A comparison of multiple machine learning algorithms on the analyzed dataset indicates considerable disparities in predicted accuracy. The proposed strategy beat all previous models, with an accuracy of 88%, demonstrating the potential benefits of using tailored or hybrid approaches. Support Vector Machine and Naïve Bayes achieved 87% accuracy, demonstrating their robustness and effectiveness in classification applications. Algorithms such as Logistic Regression, K-Nearest Neighbors, MLP Neural Networks, and LightGBM demonstrated moderate accuracy; however, classic methods such as Decision Tree and Gradient Boosting lagged, potentially because of their sensitivity to data features or inadequate parameter optimization.

These results show the importance of carefully selecting machine learning models suited to the problem context and data properties. Moreover, they highlight that ensemble or novel hybrid techniques can enhance predictive performance beyond individual standard algorithms. Future research should focus on improving these approaches through enhanced feature engineering, hyperparameter tweaking, and the use of deep learning algorithms to gain greater accuracy and generalizability across varied datasets.

Future work

The study should focus on further refining these approaches through advanced feature engineering, hyperparameter tuning, and incorporation of deep learning methods to achieve greater accuracy and generalizability across diverse datasets.

Ethics approval

Not applicable.

Availability of data and material

Not applicable.

Conflict of interest

The author declared no potential conflicts of interest concerning this article's research, authorship, and publication.

Funding

The author received no financial support for the research.

Acknowledgment

The author thanks anonymous reviewers for their comments that enhance the paper's present form.

References

[1] World Health Organization. (2024). Cardiovascular diseases (CVDs) fact sheet. World Health Organization. Retrieved August 3, 2025, from https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[2] E.J. Benjamin, P. Muntner, A. Alonso, M.S. Bittencourt, C.W. Callaway, A.P. Carson, A.M. Chamberlain, A.R. Chang, S. Chang, S.R. Das, F.N. Delling, L. Djousse, ……., Heart disease and stroke statistics—2019 update: A report from the American Heart Association. Circulation, **139**(10), e56-e528 (2019)

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

[4] K. AL-Jammali, Prediction of heart diseases using data mining algorithms, Informatica, **47**(5), 57-62 (2023).

[5] H.B.F. David and S.A. Belcy, Heart disease prediction using data mining techniques, International Journal of Scientific & Engineering Research, **9**(1), 1817-1823 (2018).

[6] A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano, Heart disease [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C52P4X (1989).

[7] K. Govindaraju and G. Kalimuthu, Heart disease analysis and prediction with machine learning techniques using cleveland dataset, In International Symposium on Intelligent Computing Systems (pp. 20-43). Cham: Springer Nature Switzerland (2024)

[8] J.D. Kelleher, M. Mac-Carthy, and O. Korvir, Fundamentals of machine learning for predictive data analytics: Algorithms, worked examples, and case studies. MIT Press (2015).

[9] R.S. Sutton and A.G. Barto, Reinforcement learning: An introduction (2nd ed.). MIT Press (2018).

[10] J. Brownlee, Machine learning mastery with Python: Understand your data, create accurate models, and work projects end-to-end. Machine Learning Mastery, Jason Brownlee, San Francisco (2016).

[11] F.Y. Osisanwo, J.E.T. Akinsola, O. Awodele, J.O. Hinmikaiye, O. Olakanmi, and J. Akinjobi, Supervised machine learning algorithms: Classification and comparison, International Journal of Computer Trends and Technology (IJCTT), **48**(3), 128–138 (2017).

[12] D. Shah, S. Patel, and S.K. Bharti, Heart disease prediction using machine learning techniques, SN Computer Science, **1**(6), 345 (2020).

[13] M. Fatima and M. Pasha, Survey of machine learning algorithms for disease diagnostic, Journal of Intelligent Learning Systems and Applications, **9**(01), 1-16 (2017).

[14] K. Pahwa and R. Kumar, Prediction of heart disease using hybrid technique for selecting features, In: 2017 4th IEEE Uttar Pradesh section international conference on electrical, computer, and electronics (UPCON). IEEE, 500–504 (2017).

[15] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia, and J. Gutierrez, A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease, In: 2017 IEEE symposium (2017).

[16] T. Liu, A. Krentz, L. Lu, and V. Curcin, Machine learning based prediction models for cardiovascular disease risk using electronic health records data:

systematic review and meta-analysis, European Heart Journal-Digital Health, **6**(1), 7-22 (2025).

[17] R. Chauhan, P. Bajaj, K. Choudhary, Y. Gigras, Framework to predict health diseases using attribute selection mechanism, In: 2015 2nd international conference on computing for sustainable global development (INDIACom). IEEE, p. 1880–84 (2015).

[18] H. Bouali, J. Akaichi, Comparative study of different classification techniques: heart disease use case, In: 2014 13th international conference on machine learning and applications. IEEE, p. 482–86 (2014).

[19] S. Xu, Z. Zhang, D. Wang, J. Hu, X. Duan, and T. Zhu, Cardiovascular risk prediction method based on CFS subset evaluation and random forest classification framework. In: 2017 IEEE 2nd international conference on big data analysis (ICBDA), IEEE, p. 228–32 (2017).

[20] D.W. Hosmer, S. Lemeshow, and R.X. Sturdivant, Applied Logistic Regression (3rd ed.). Wiley (2013).

[21] C. Cortes and V. Vapnik, Support-vector networks, Machine Learning, **20**(3), 273–297 (1995).

[22] J.H. Friedman, Greedy function approximation: A gradient boosting machine. The Annals of Statistics, **29**(5), 1189–1232 (2001).

[23] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, Learning representations by back-propagating errors, Nature, **323**(6088), 533–536 (1986).

[24] T. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794 (2016).

[25] M. Niazkar, A. Menapace, B. Brentan, R. Piraei, D. Jimenez, P. Dhawan, and M. Righetti, Applications of XGBoost in water resources engineering: A systematic literature review (Dec 2018–May 2023). Environmental Modelling & Software, **174**, 105971 (2023).

[26] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Liu, LightGBM: A highly efficient gradient boosting decision tree, Conference Advances in Neural Information Processing Systems, 30 (2017).

[27] D.N. Wang, L. Li, and D. Zhao, Corporate finance risk prediction based on LightGBM. Information Sciences, **602**, 259-268 (2022).