

Heart Disease Detection using Machine Learning Algorithms (XGBoost, Random Forest, KNN, SHAP)

Sifa Sasture, S.A. Gaikwad

M-Tech Student Dept. of Computer Science & Engineering College of Engg. Osmanabad Professor,
Dept. of Computer Science & Engineering, College of Engg. Osmanabad

ABSTRACT: -Predicting critical health conditions in their early stages can make the difference between life and death, and one such health condition is heart disease. Over the last decade, the main reason for death has been heart disease. Heart Disease is an ailment that affects many lives, is severely life-threatening, and can impair a person's ability to live a conventional life. The delay in treating Heart Disease increases the endangerment of the afflicted person. Consequently, early diagnosis of it can help save countless lives. However, the reasons for Heart Disease are varied, making its prediction very complex. Our objective is to use Machine Learning to enhance the dependability and simplicity of the prediction of Heart Disease. It was concluded that three datasets should be used; two have an immense size, alongside many Machine Learning algorithms. The proposed algorithms were tested: k-Nearest Neighbour, Gradient Boosting, Random Forest, Naïve Bayes, Decision Tree, and Logistic Regression. After rigorous testing, the only algorithm, Logistic Regression, stayed dominant in most of the testing achieving accuracies of 91.6% and 90.8%. Still, on the last dataset, the best algorithm was a random forest which scored the highest accuracy in all the testing, 98.6%. As shown in this paper, Machine Learning is a superb approach to predicting Heart Disease, and results can be further improved with the help of medical professionals and more research.

Keywords: Heart Disease prediction Machine Learning Classification Naïve Bayes Gradient Boosting Linear Regression K-Nearest Neighbor.

INTRODUCTION

Cardiovascular diseases (CVDs) are the leading cause of death globally, accounting for nearly 17.9 million deaths each year according to the World Health Organization [1]. Among these, heart disease is one of the most critical conditions that requires early detection and intervention to prevent severe complications and fatalities. Traditional methods of diagnosis, such as Electrocardiograms (ECG), angiography, and stress testing, while accurate, are often time-consuming, costly, and require expert medical practitioners [1]. With the rapid growth of Artificial Intelligence (AI) and data-driven techniques, Machine Learning (ML) has emerged as a powerful tool in the healthcare domain [2]. ML algorithms have the ability to analyze large volumes of medical data, identify hidden patterns, and provide accurate disease predictions, assisting doctors in clinical decision-making [2]. In particular, ensemble methods such as Random Forest and gradient boosting have shown promising results in medical diagnosis tasks, including heart disease prediction[2].

However, one of the major limitations of traditional ML models is their “black-box” nature, which makes them difficult to interpret in medical applications [3]. Doctors and healthcare professionals often require not only predictions but also explanations of how those predictions are made. To address this challenge, SHAP (SHapley Additive exPlanations), a state-of-the-art explainable AI technique, is integrated into this project [3]. SHAP provides feature-level interpretability by quantifying the contribution of each input

factor—such as age, cholesterol, blood pressure, or chest pain type—towards the final prediction [3]. This project aims to build a predictive system for heart disease using the UCI Heart Disease dataset, Random Forest classifier, and SHAP explainability framework [3]. The proposed system is deployed as a web-based application using Flask, enabling real-time prediction, visualization of results, and transparency through SHAP explanations [3]. This not only improves the accuracy of heart disease diagnosis but also enhances trust among healthcare practitioners by providing interpretable insights into model behavior [3].

Methods

First, we gathered and **preprocessed** the dataset to remove any necessary inconsistencies, such as replacing null occurrences with average values [1]. We divided the dataset into two distinct groups, named the test dataset and the training dataset, respectively. Next, we implemented several distinct classification algorithms to determine which one achieved the highest accuracy for these datasets [1].

proposed methodology :-This study investigates ML techniques such as Naive Bayes, SVM, Voting, XGBoost, Random Forest, K-Nearest Neighbors (KNN), Decision Tree (DT), and Logistic Regression (LR) classifiers [2]. These algorithms can aid doctors and data analysts in making correct diagnoses of cardiac disease. This article incorporates recent data on cardiovascular illness, as well as relevant journals, research, and publications [2]. The methodology, as in [3], provides a framework for the suggested model. The methodology is a set of steps that transform raw data into consumable and identifiable data patterns. *The proposed approach consists of three stages:*

- The first stage is data collection;
- The second stage extracts specific feature values;
- The third stage is data exploration, as shown in Figure 1.

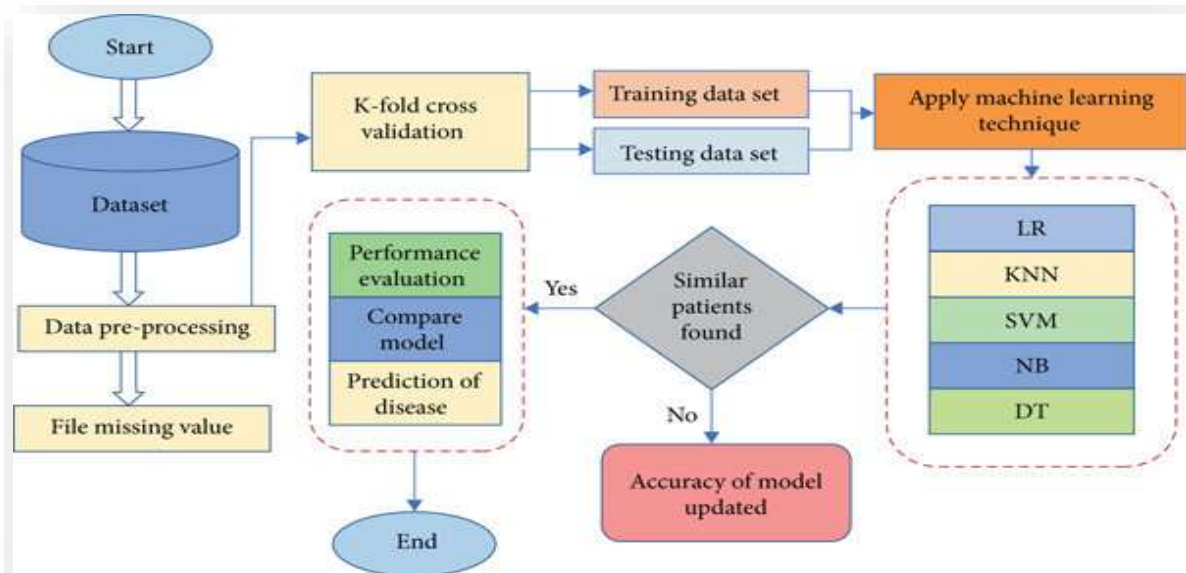


Fig 1: Workflow of Heart Disease Prediction using Machine Learning Algorithms

Feature no.	Feature name	Feature code	Description	Values type
1	Age	AGE	Age of patient	Number of years
2	Gender	GEN	Patient sex	Female = 0, male = 1
3	Chol	CHOL	Evaluation of a patient's cholesterol levels	mg/dl
4	Trestbps	HRP	Blood resting pressure	Mm
5	CP	CPT	Chest pain types	Typical angina = 1, atypical angina = 2, nonanginal pain = 3, asymptomatic = 4
6	Fbs	FBS	Blood sugar in fasting case	< or > 120 mg/dl (true = 1, false = 0)
7	Thalach	MHR	Maximum rate achieved on heart	Continuous
8	RestEcg	REC	Electrocardiograph by resting	0 = no abnormalities, 1 = normal, 2 = left ventricular hypertrophy (possible or certain)
9	Oldpeak	OP	ST depression when compared to rest taken quantity	Continuous
10	Exang	EIA	Angina caused by exercise	1 = there is pain, 0 = there is no pain
11	Ca	CMV	Count of main vessels colored by fluoroscopy	0-3
12	Slope	PES	Peak exercise ST segment slope	Up sloping = 0, flat = 1, down = 2
13	Thal	TS	Thallium stress	Negative = 0, positive = 1, inconclusive = 2
14	Target		target variable representing diagnosis of heart disease using the angiographic disease status.	0 = no heart disease (< 50% diameter narrowing) 1 = heart disease (> 50% diameter narrowing)

Table 2. The used features from the CHDD.

The dataset was processed to handle missing values, cleanse the data, and perform normalization [4]. We then classified the pre-processed data using the ten classifiers (A1, A2, ..., A10). Finally, after putting the suggested model into practice, we evaluated its performance and accuracy using a range of performance measures [5]. This model developed a Reliable Prediction System for Heart Disease (RPSHD) using a variety of classifiers. The model uses 13 medical factors for prediction, including age, sex, cholesterol, blood pressure, and electrocardiography [3].

Datasets and dataset features

This research employs both the CHDD and a private dataset for heart disease prediction. The CHDD dataset has 303 samples, while the private dataset has 200, and they have the same features. The combined dataset contains 503 records, and 13 features are associated with each one (including demographic, clinical, and laboratory parameters) [6]. The datasets have many features that can be used for heart disease prediction including age, gender, blood pressure, cholesterol levels, electrocardiogram readings (ECG), chest pain, exercise-induced angina, blood sugar with fasting condition, max heart rate achieved, oldpeak, coronary artery, thalassemia, and other clinical and laboratory measurements, as shown in Table 2. The outcome variable known as "Target" takes a binary value and refers to the heart disease predicting feature (i.e., it indicates whether or not cardiac disease is present). Figure 2 shows the percentage distribution of individuals with heart disease in the combined datasets. A total of 503 samples have been gathered, and 45.9% of those have been diagnosed with HD, while the remaining 54.1% of individuals have not been infected with the disease. Boxplots are an effective visualization technique for understanding the distribution of data and identifying potential outliers. By applying boxplots to a dataset related to HD, one can get insights into the distribution of a variety of HD-related features or variables.

The HD dataset's are Boxplots are illustrated in Fig. 3. Boxplots are used to illustrate the distribution of scores for HD detection in this figure [8]. Every graph we obtained had an anomaly. Removing them will cause the median of the data to drop, which might make it harder to detect HD accurately.

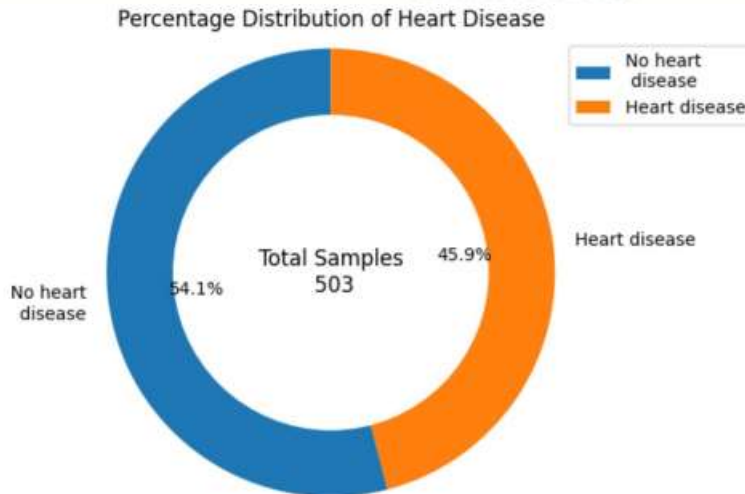


Fig. 2. The percentage distribution of heart disease in the Combined dataset.

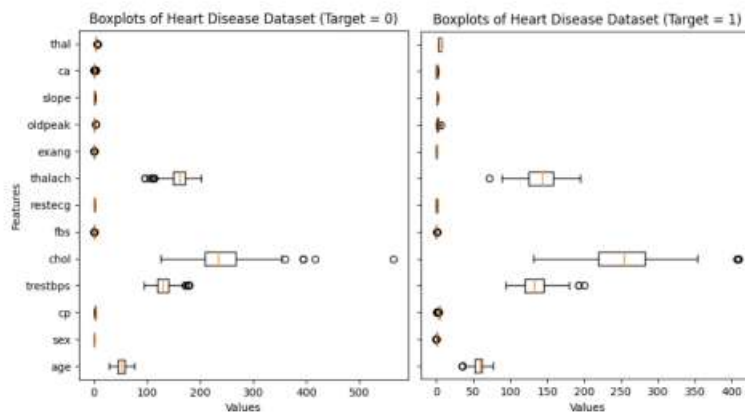


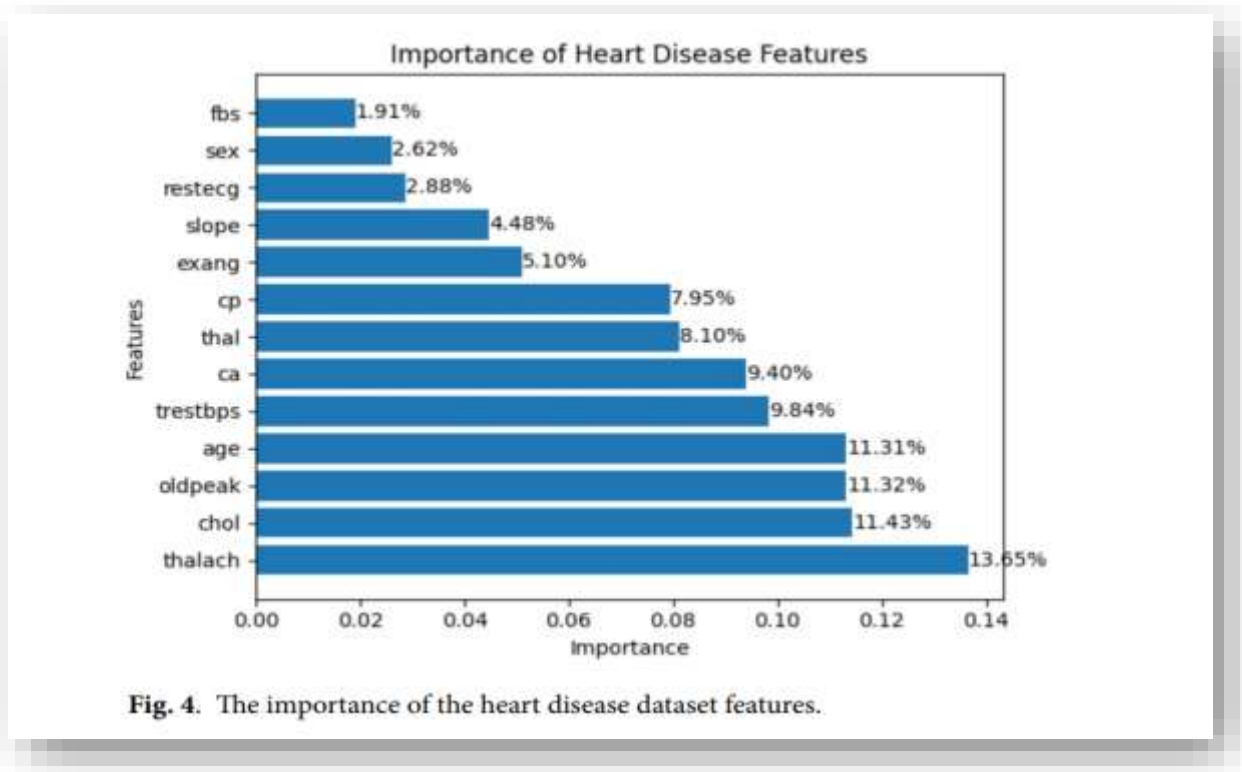
Fig. 3. Boxplots of the combined heart disease dataset.

On the other hand, this method offers more benefits than the others; by identifying heart disease infection at an early stage, when medical care is most beneficial, this diagnostic could preserve lives.

Feature selection

In this research, we perform feature selection and classification using the Scikit-learn module of Python [7]. Initially, the processed dataset was analyzed using several different ML classifiers, including RF, LR, KNN, bagging, DT, AdaBoost, XGBoost, SVM, voting, and Naive Bayes, which were evaluated for their overall accuracy. In the second step, we used the Seaborn libraries from Python to create heat maps of correlation matrices and other visualizations of correlations between different sets of data [8].

Heart disease dataset features are crucial for early and accurate prediction of cardiac issues, enabling preventative strategies, personalized treatments, and improved patient outcomes. Key features include age, sex, chest pain type, cholesterol levels, blood pressure, exercise-induced angina, and electrocardiogram result, when analysed through data science techniques, reveal complex patterns and risk factors associated with heart conditions [6][7].



The use of SHAP methods

SHAP (SHapley Additive exPlanations) is a powerful tool in machine learning that helps interpret and explain the predictions made by complex models [9]. The following are the benefits that SHAP offers in ML applications:

1. **Enhanced Transparency:** SHAP makes black-box models more transparent, fostering trust among users and stakeholders. This is especially crucial in industries like finance, healthcare, and legal, where understanding model decisions is essential.
2. **Regulatory Compliance:** Many industries are subject to regulations that require model decisions to be explainable. SHAP ensures compliance by providing clear, understandable explanations for each decision, facilitating documentation, and sharing with regulators.
3. **Improved User Trust and Adoption:** When end users understand why a model is making certain predictions, they are more likely to trust and adopt the technology. User interfaces can incorporate SHAP explanations to improve the user-friendliness of AI-powered applications [9].
4. **Actionable Insights:** SHAP doesn't just explain predictions; it also provides actionable insights. For example, in prediction models, SHAP can identify key contributing features, allowing doctors to take proactive steps to detect disease earlier [10].

5. **Facilitates Collaboration:** SHAP explanations can bridge the gap between data scientists and non-technical stakeholders, facilitating better communication and collaboration. By providing a common understanding of model behavior, teams can work more effectively together [10].

```

#----- SHAP Summary Plot (Random Forest) -----#
print("🔍 Generating SHAP plot for Random Forest...")

# Assuming model = RandomForestClassifier(...)
import shap
import matplotlib.pyplot as plt

# Use original (unscaled) features for SHAP
X_shap = X_test # IF not defined already

# Get SHAP values
explainer = shap.TreeExplainer(model)
shap_values = explainer.shap_values(X_shap)
plt.figure(figsize=(8, 3))
# Clear any previous plot
plt.clf()

# Check if SHAP returns list of values (e.g., for binary classifier)
if isinstance(shap_values, list):
    print(f"SHAP list length: {len(shap_values)}") # typically 2 for binary
    print(f"Shape of shap_values[1]: {shap_values[1].shape}")
    print(f"Shape of X_shap: {X_shap.shape}")

    # 🟢 Use shap_values[1] for class 1
    shap.summary_plot(shap_values[1], X_shap, show=False)
else:
    # Newer SHAP versions may return Explanation object directly
    shap.summary_plot(shap_values, X_shap, show=False)

# Save the plot
plt.savefig("static/shap_summary_rf.png", bbox_inches="tight", dpi=150)
print("📄 SHAP summary plot saved to static/shap_summary_rf.png")

print("X_shap shape:", X_shap.shape)
print("shap values[1] shape:", shap_values[1].shape)

```

Fig 5. Shap Summary Plot

SHAP dependence plot

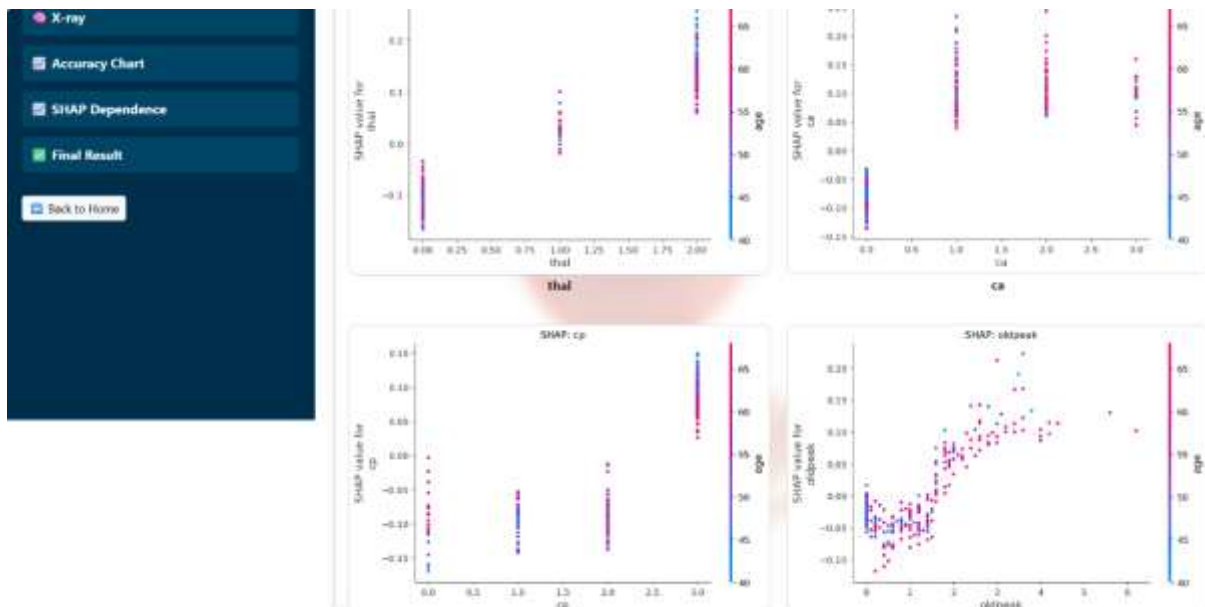


Fig 6. SHAP Dependence Plots for Selected Features (thal, ca, ...)

Accuracy Comparison

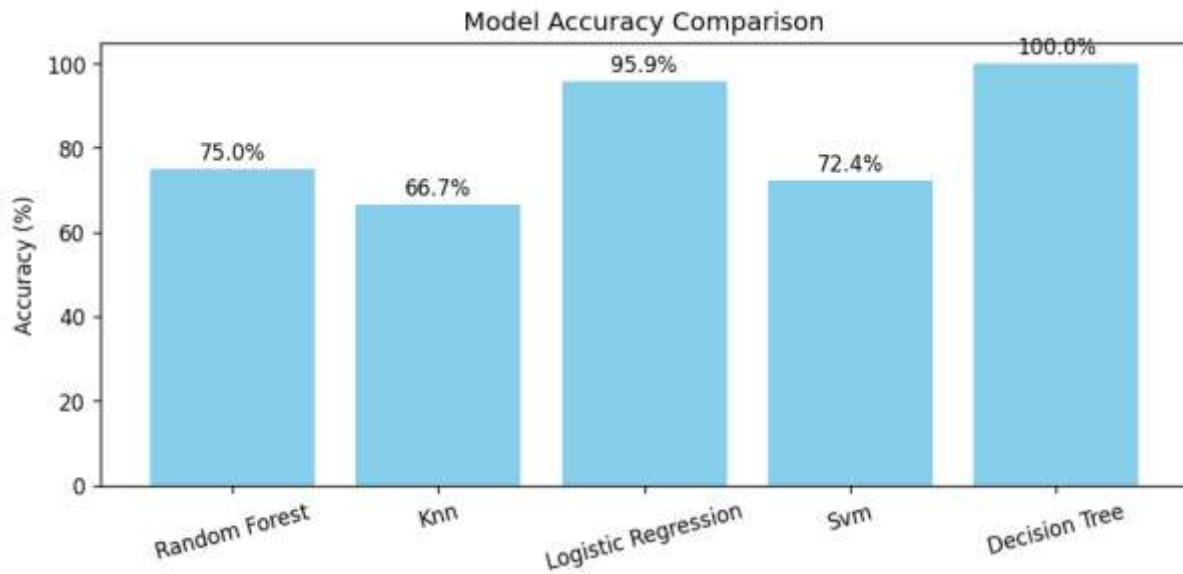


Fig 7.The accuracy results of the ML Algorithms.

In this project on **Heart Disease Prediction**, multiple machine learning algorithms were implemented, trained, and evaluated on the heart disease dataset. The main models considered were **Random Forest (RF)**, **K-Nearest Neighbors (KNN)**, and **Extreme Gradient Boosting (XGBoost)**. Their performance was analyzed and compared based on accuracy and predictive capability.

1.1 Decision Tree: The supervised learning type includes the decision tree algorithm. Both regression and classification issues may be handled using them. Each node in the tree corresponds to a class label, with attributes expressed on the tree's inner node [11].

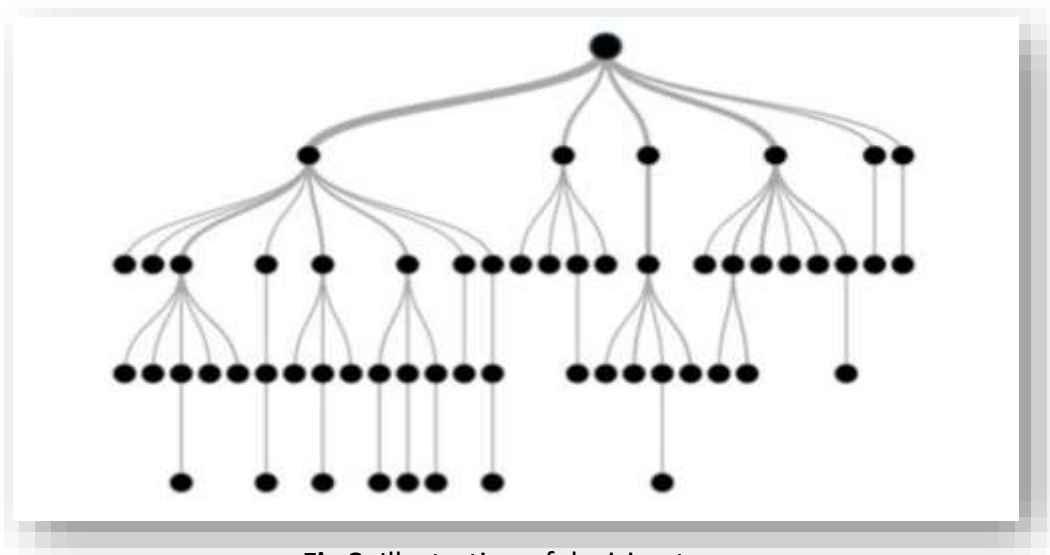


Fig 8. Illustration of decision tree

The entropy varies when a node is employed in a decision tree, and it breaks down the training dataset into smaller groupings. The information denotes the increase in entropy [11].

Definition: Suppose S is a set of instances, A is an attribute, S_v is the subset of S with $A = v$, and $Values(A)$ is the set of all possible values of A , then

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

1.2 K-Nearest Neighbor

Hodges and fix established a classification of nonparametric pattern algorithm described as the (KNN) K-Nearest Neighbor rule in 1951. [12]. The KNN technique is one of the best basic and most powerful classification methods. It doesn't make any assumptions about data and is classification usable jobs where very little or no prior knowledge of the distribution of data is access able. This algorithm is used to find The value of the found data points in it is allocated to the nearest data points in the training set to the data point for which a target value is assigned.

1.3 Logistic Regression

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique[13]. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

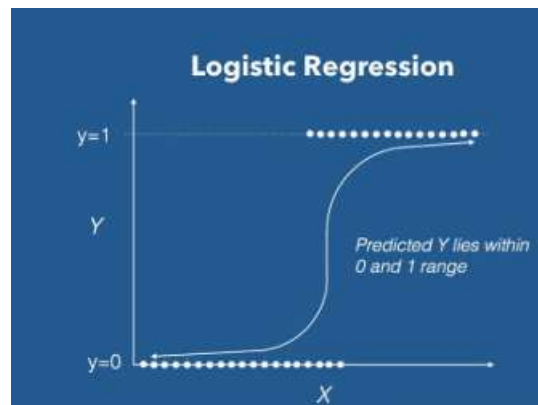


Fig 9. Logistic Regression

Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1). The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc. Logistic Regression is a significant machine learning

algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

1.4 Random Forest

Random Forest is one of the supervised machine learning algorithms that can be used for classification and regression tasks but works better in classification tasks. This algorithm considers

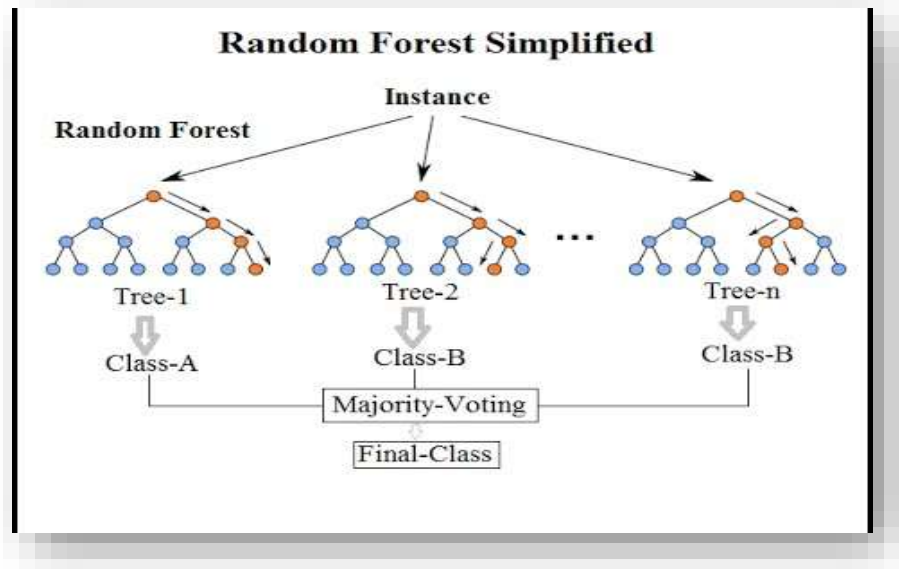


Fig 10. Random forest demonstration

multiple decision trees before giving an output. It employs a voting approach for classification and then determines the class, whereas it uses the mean of all the decision tree outputs for regression[14]. Random Forest Algorithm is extremely efficient with large datasets with high dimensionality.

1.5 XGBoost (Extreme Gradient Boosting)

XGBoost is an advanced implementation of the gradient boosting algorithm designed for high speed and performance [15]. In this **Heart Disease Prediction** project, XGBoost was used to classify patients based on clinical features such as age, cholesterol, and blood pressure. It builds multiple decision trees sequentially, where each tree corrects the errors of the previous one, improving overall accuracy. XGBoost includes **regularization** to prevent overfitting and provides better generalization on unseen medical data. Among all the models tested, it achieved the **highest prediction accuracy** for heart disease detection.

CONCLUSION

An early diagnosis is crucial to saving as many lives as possible, and Machine learning proved to be a great approach to detect this cunning disease prematurely. Logistic regression excelled in predicting heart disease in most datasets with accuracies of 91.6%, and 90.8%, but it was beaten in the last dataset only by Random Forest which had an accuracy of 98.6%. With more research and guidance from medical professionals, prediction accuracy can grow even more. Machine Learning can be applied to many fields, not just medicine, and it can be used to predict anything from stock prices to the results of sports matches, making it a very useful tool for humanity. And this tool will only keep improving and producing better results.

REFERENCES

- [1] World Health Organization (WHO), "Cardiovascular Diseases (CVDs) Key Facts," WHO, Geneva, 2024.
- [2] S. U. Amin, M. S. Hossain, and M. R. Hossain, "Machine Learning Based Disease Diagnosis: A Comprehensive Review," *Journal of Healthcare Engineering*, vol. 2019, Article ID 2703492, 2019.
- [3] J. Brownlee, *Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models, and Work Projects End-to-End*, Machine Learning Mastery, 2020.
- [4] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed., O'Reilly Media, 2019.
- [5] 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 (KDD)*, 2016.
- [6] UCI Machine Learning Repository, "Heart Disease Dataset," University of California, Irvine, 2024.
- [7] F. Pedregosa, G. Varoquaux, A. Gramfort, et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [8] M. Waskom, "Seaborn: Statistical Data Visualization," *Journal of Open Source Software*, vol. 6, no. 60, p. 3021, 2021.
- [9] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [10] M. Halloran, "Explainable AI and SHAP Values in Healthcare: Building Trust in Machine Learning Models," *Journal of Biomedical Informatics*, vol. 139, 104317, 2023.
- [11] J. R. Quinlan, "Induction of Decision Trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [12] E. Fix and J. L. Hodges, "Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties," *Technical Report 4, USAF School of Aviation Medicine*, Randolph Field, Texas, 1951.
- [13] D. R. Cox, "The Regression Analysis of Binary Sequences," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 20, no. 2, pp. 215–242, 1958.



- [14] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [15] 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*, pp. 785–794, 2016.