# Using machine learning to Predict Heart Disease

NIKHIL BORA, SREEDEVI GUTTA, AHMAD HADAEGH
Department of Computer Science and Information System
California State University San Marcos
333 Twin Oak valley Rd. San Marcos CA, 92009
UNITED STATES OF AMERICA

*Abstract*— Heart Disease has become one of the most leading cause of the death on the planet and it has become most life-threatening disease. The early prediction of the heart disease will help in reducing death rate. Predicting Heart Disease has become one of the most difficult challenges in the medical sector in recent years. As per recent statistics, about one person dies from heart disease every minute. In the realm of healthcare, a massive amount of data was discovered for which the data-science is critical for analyzing this massive amount of data. This paper proposes heart disease prediction using different machine-learning algorithms like logistic regression, naïve bayes, support vector machine, k nearest neighbor (KNN), random forest, extreme gradient boost, etc. These machine learning algorithm techniques we used to predict likelihood of person getting heart disease on the basis of features (such as cholesterol, blood pressure, age, sex, etc. which were extracted from the datasets. In our research we used two separate datasets. The first heart disease dataset we used was collected from very famous UCI machine learning repository which has 303 record instances with 14 different attributes (13 features and one target) and the second dataset that we used was collected from Kaggle website which contained 1190 patient's record instances with 11 features and one target. This dataset is a combination of 5 popular datasets for heart disease. This study compares the accuracy of various machine learning techniques. In our research, for the first dataset we got the highest accuracy of 92% by Support Vector Machine (SVM). And for the second dataset, Random Forest gave us the highest accuracy of 94.12%. Then, we combined both the datasets which we used in our research for which we got the highest accuracy of 93.31% using Random Forest.

*Keywords*— Heart Disease, Machine learning, naïve Bayes, logistic regression, support vector machine, KNN, random forest, extreme gradient boost

Received: March 9, 2021. Revised: November 10, 2021. Accepted: December 2, 2021. Published: January 3, 2022.

## 1. Introduction

The most crucial part of our body is Heart. It's a muscular organ which placed directly behind and slightly left of breastbone. Heart Disease causes highest number of deaths globally, with approximately around 17.9 million people died from it every year which means around 31% of deaths are from the heart disease as per the WHO (World Health Organization). Heart disease are also called as Cardiovascular Disease which are group of complication of the blood vessels and heart which include cerebrovascular disease, rheumatic heart disease and some other heart conditions. Four out of five heart disease deaths are from the strokes and heart attacks. Heart Disease is the most life-threatening disease in the world nowadays. Therefore, heart disease should be predicted at their early stage and healthy lifestyle are ways to prevent them.

People who are at risk of heart disease may have symptoms like high blood pressure, obesity, cholesterol, diabetes, age, etc. As there is recent improvement in medical health care is observed. The health care system has collected massive amount of data about heart disease and they have all those data and created datasets which consist of different medical parameter or features such as age, sex, blood pressure, cholesterol, chest type and so on, etc. Datasets consist of around 13 to 15 different medical parameters. These datasets are now available for analysis and to extract crucial information from it. So, we can predict the heart disease at their early stage by applying machine learning algorithms on this massive amount of data to extract features (information/medical parameters) that we will extract from datasets. Various machine learning techniques like logistic regression, naïve-Bayes, support vector machine, k nearest neighbor (KNN), etc. we can use for predicting heart disease means to classify whether a person is having cardiovascular disease or not, after applying them on the feature's extraction from datasets. Different algorithms will give different accuracy, so after comparing among them we can find the best algorithm which predicts heart disease with highest

accuracy. The main objective of our project is to enhance efficiency for predicting heart disease rate.

## 2. Related Work

Heart disease is the leading cause of death nowadays. In the work of Umair Shafique *et. al. [*1], the authors used data mining techniques, decision tree, Naïve Bayes and Neural Network algorithms, for which they got accuracy of 82% for Naïve Bayes and of 78% for Decision tree. Authors used WEKA (https://sourceforge.net/projects/weka/) machine learning software in their work. In the work of Sabarinathan Vachiravel *et. al.* [2], the authors proposed a decision machine learning algorithm to predict heart disease and achieved 85% accuracy using decision tree.

In the work of Vikas Chaurasia et. al. [3], the dataset the authors used in their work was downloaded from UCI laboratory which has 14 different attributes out of which they only used 11 attributes to predict heart disease using Naïve Bayes and Decision Tree machine learning algorithms. WEKA tool was used in their work, using which they achieved 82% accuracy for Naïve Bayes, and 84% for decision tree. In the work of N. Komal Kumar, G. Sarika Sindhu et. al. [4], the authors proposed machine learning algorithms such as Random Forest, Logistic Regression, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) for heart disease prediction. The highest accuracy they achieved is 85% using Random Forest algorithm, 74% for Logistic Regression, 77% for SVM. The lowest accuracy they got is using K-Nearest Neighbors (KNN) of 68%. The dataset they used in their work was unbalanced, resulting in a need for applying sampling techniques. But they directly applied machine learning algorithms without filtering data in the dataset.

Malkari Bhargav et. al. [5], proposed to identify heart disease using different ML techniques. They collected dataset from UCI ML repository. Dataset has total of 14 parameters like age, blood pressure, cholesterol. He achieved highest accuracy of 96% using ANN. 88% using Logistic regression, 83% using Random Forest, Decision Tree 83%, 70% using SVM, and the lowest accuracy he got is 68% using KNN. Gayatri Ramamoorthy et. al. [6] made use to forecast heart disease using ML models. The authors got highest accuracy score for KNN 83% and lowest accuracy score for SVM of 65% and for Naïve Bayes 80%. Apurb Rajdhan et. al. [7] used ML algorithms like decision tree, logistic regression, random forest and naïve Bayes used to analyze cardiovascular disease and achieved accuracy like 81%, 85%, 90%, and 85% respectively. Hana H. Alalawi et. al. [8] used deep learning and machine algorithms to diagnose the heart disease using combination of two datasets which was collected from Kaggle and Cleveland dataset for heart. Using which he achieved maximum accuracy using Random Forest 92%. For Naïve Bayes 83%, ANN 77%, KNN 71%, Logistic regression 75%, SVM 72% respective accuracies he got. J. Maiga et. al. [9] has developed model for predicting heart disease which utilizes various combination of features. Various classification algorithms were used which are KNN, naïve Bayes and random forest. The authors achieved highest accuracy of 73% using Random Forests. They didn't achieve good accuracies because they didn't perform feature scaling and normalization on data. A. Lakshmanrao et. al. [10] used several data mining and gradient boosting algorithms in their research for diagnosing heart disease. The authors applied several sampling techniques for handling unbalanced datasets. Dataset called "Framingham heart disease" was collected from Kaggle. Dataset has 15 features and total of 4220 patient records. They used several boosting algorithms like Adaboost and Gradient boosting for which they achieved accuracy of 78% and 88% respectively. Still for Naïve Bayes they got the lowest accuracy of just 61% and for logistic regression 66%. Ashok Kumar Dwivedi et. al. [11] evaluated performance of machine learning techniques for predicting heart disease using ten-fold cross validation naïve Bayes, KNN, ANN, SVM, and Logistic Regression for which accuracies they achieved is 83%, 80%, 84%, 82%, and 85% respectively.

Muhammad Saqlain et al.[12] identified heart failure using unstructured data of heart patients. In their work they used several ML algorithms, and their accuracies are Logistic Regression 80%, SVM 83%, Random Forest 86% and Decision tree 86% and neural network 84%. Hossam Meshref et. al. [13] compared several ML algorithms like SVM, Naïve Bayes, MLP and also did selected specific features from the datasets for which they got different accuracy when selecting different features like when they selected all 14 features of datasets, they got highest accuracy using naïve bayes of 81% but when they selected only specific features then they got highest accuracy using SVM. From their research, they proved that feature selection is the most important step for improving accuracy of

machine learning algorithms. Baban Rindhe et. al. [14] compared 3 data mining classifier techniques such as Support vector classifier, Random forest classifier and neural network among all these 3, Support vector classifier proved to be effective as it achieved the highest accuracy of 84% for predicting heart disease.

H. Jayashree et. al. [15] did research on heart disease diagnosis in which the authors used both gradient boosting as well as classification algorithms like SVM, Naïve bayes, AdaBoost, GradientBoost algorithm among which they achieved maximum accuracy using Naïve bayes of 85%. For AdaBoost she got 82% and for gradient boosting they got 83% accuracy and SVM 84%. They also used an Extra tree classifier for which she got 81% accuracy.

In comparison to the previous works, we used combination of datasets in our research and got very high accuracy.

## 3. Methodology

The main goal of our proposed method is to predict the likelihood of person having the chance of getting heart disease based on a person's medical parameters (cholesterol, blood pressure, age, etc.), so that disease may be detected early and effectively. In this research, we have used 6 various machine learning algorithms:

1. Logistic Regression
2. Support Vector Machine (SVM)
3. Random Forest
4. Naïve Bayes
5. K-nearest neighbors (KNN)
6. Gradient Boost

We will compare the above 6 models to find out which will give the best accuracy for prediction of heart disease.

The language that I used in this research is Python. Because of its simplicity and flexibility, easy to understand code, access to huge amounts of frameworks and libraries like pandas numpy, it's widely popular for machine learning. One more reason why it's used in machine learning is because it offers lots of visualization tools which makes understanding data much easier like matplotlib which is the most used library for data visualization as it enables us to generate histograms, plots and charts for representing data. All these reasons make python perfectly fit for machine learning projects. In this project, I used anaconda navigator's Jupyter notebook for analyzing the data. Using Jupyter notebook we can write code, especially for performing visualization and plotting graphs and processing the data.

Table 1: Dataset 1: Heart disease dataset collected from UCI machine learning repo.

| Sr. no. | Parameter/ attribute | Info. |
|---|---|---|
| 1 | age | Patient's age in years |
| 2 | sex | 0 = female ; 1 = male [categorical variable] |
| 3 | cp | Chest pain type [categorical variable]<br>0 = typical angina<br>1 = atypical angina<br>2 = non-anginal pain<br>3 = asymptomatic |
| 4 | trestbps | Resting blood pressure (mm hg) |
| 5 | chol | Cholesterol (mg/dl) |
| 6 | restecg | Resting electrographic results [categorical variable]<br>0 = normal<br>1 = having ST_T_wave abnormality |
| 7 | fbs | Fasting blood sugar<br>0 = less than 120 mg/dl<br>1 = more than 120 mg/dl |
| 8 | thalach | Maximum heart rate achieved |
| 9 | exang | Exercise induced angina [categorical variable]<br>0 = no<br>1 = yes |
| 10 | oldpeak | Exercise induced ST depression in comparison with rest state |
| 11 | slope | Slope of exercise ST segment [categorical variable]<br>0 = unslope<br>1 = flat<br>2 = downslope |
| 12 | ca | No. of major vessels[0-3] colored by fluoroscopy |
| 13 | thal | Defect type [categorical variable]<br>3 = normal<br>6 = fixed<br>7 = reversible defect |
| 14 | target | Has heart disease or not<br>0 = no<br>1 = yes |

The dataset contains 14 attributes out of which 13 are features and one target attribute. It has 303 rows and 14 columns. So the dataset's shape is (303,14). All 14 attributes are Shown in table 1.

Table 2 : Dataset 2: Heart disease Dataset (Comprehensive) collected from Kaggle

| Sr. no. | Parameter/ attribute | Info. |
|---|---|---|
| 1 | Age | Patient's age in years |
| 2 | Sex | 0 = Female ; 1 = Male [categorical variable] |
| 3 | chest pain type | Chest pain type [categorical variable]<br>1 = Typical angina<br>2 = Atypical angina<br>3 = Non-anginal pain<br>4 = Asymptomatic |
| 4 | resting bps | Resting blood pressure (mm hg) |
| 5 | Cholesterol | Serum cholesterol in mg/dl |
| 6 | fasting blood sugar | Fasting blood sugar<br>0 = Less than 120 mg/dl<br>1 = More than 120 mg/dl |
| 7 | resting ecg | Resting electrographic results [categorical variable]<br>0 = Normal<br>1 = Having ST_T_wave abnormality |
| 8 | max heart rate | Maximum heart rate achieved |
| 9 | exercise angina | Exercise induced angina [categorical variable]<br>0 = No<br>1 = Yes |
| 10 | Oldpeak | Exercise induced ST depression in comparison with rest state |
| 11 | ST slope | Slope of exercise ST segment [categorical variable]<br>0 = Normal<br>1 = Unsloping<br>2 = Flat<br>3 = Down sloping |
| 12 | Target | Has heart disease or not<br>0 = No<br>1 = Yes |

### 3.1 Datasets

We used two datasets in this research for predicting heart disease. One is collected from the famous UCI Machine learning official website and another dataset is a combination of datasets collected from Kaggle.

The first dataset that was used in this research was collected from UCI repository from the following link :

https://archive.ics.uci.edu/ml/datasets/HeartDisease

This second dataset was collected from the Kaggle site; it's the most comprehensive heart disease dataset. It's also available on IEEE-data port website. Following is the link to the dataset:

https://www.kaggle.com/sid321axn/heart-statlog-cleveland-hungary-final

Basically, this dataset is a combination of 5 most popular datasets for heart disease which are Cleveland, Long Beach, Switzerland, Hungarian and Statlog heart dataset. The dataset contains a total of 1190 records with 11 features. The shape of this dataset (1190,12).

As it contains 12 attributes out of which are 11 features and 1 target variable. All 12 attributes are as shown in Table 2.

### 3.2 Data Cleaning and Preprocessing

The datasets which were collected from UCI machine learning repository and Kaggle website contain unfiltered data which must be filtered before the final data set can be used to train the model. Also, data has some categorical variables which must be modified into numerical values for which we used Pandas library of Python. In data cleaning step, first we checked whether there are any missing or junk values in the dataset for which we used the isnull() function. Then for handling categorical variables we converted them into numerical variables by creating dummy variables for each categorical variable using get_dummies() function of Pandas library.

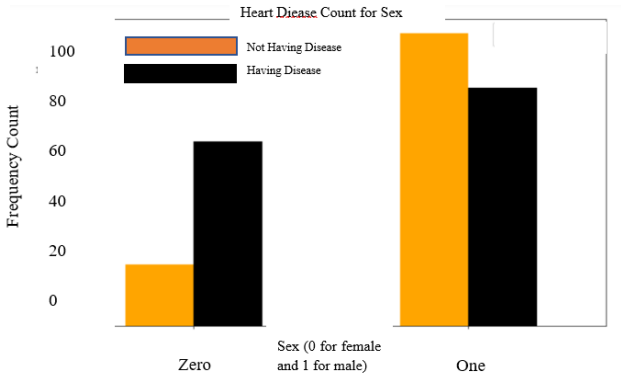


Figure 1. Heart Disease Count for Sex

In Figure. 1, we plotted Heart disease count for sex, from which we can see that Men are more prone to heart disease than females.

We also plotted Figure 2 heatmap of correlation matrix between variables for dataset 1, from which we figured out that target variable which is

dependent variable is negatively correlated to some of the factors like thalach, slope, cp and positively correlated to some of the parameters like thal, ca, etc. In simple words, heatmap is very useful for data visualization as it tells us a lot about how these 14 attributes of datasets are correlated to each other.

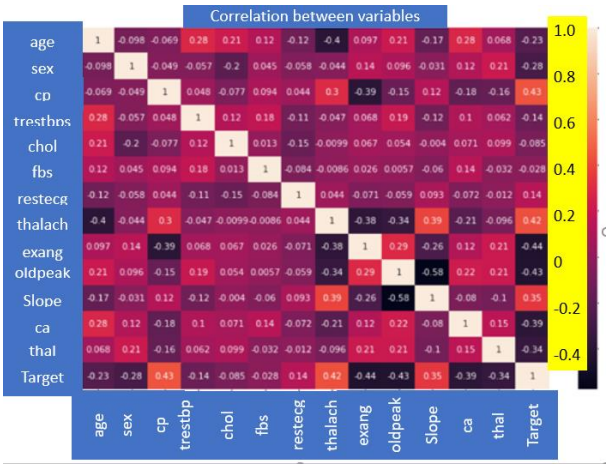


Figure 2: Plotted heatmap of correlation matrix (Dataset 1)

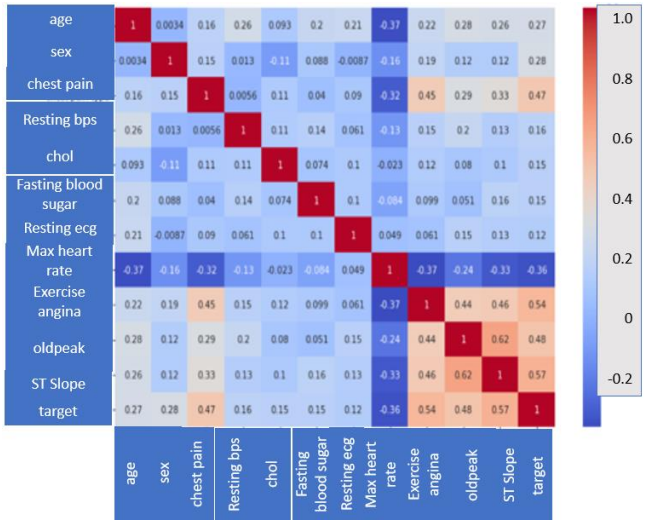


Figure 3: Plotted heatmap of correlation matrix (Dataset2)

Similarly, for dataset 2, in Figure 3. we can see the correlation of the different features with the target variable.

## 3.3 Machine Learning Algorithms

### 3.3.1 Random Forest :

Random Forest is the most famous and it is considered as the best algorithm for machine learning. It is a supervised learning algorithm. To achieve more accurate and consistent prediction, random forest creates several decision trees and combines them together

The major benefit of using it is its ability to solve both regression and classification issues. When building each individual tree, it employs bagging and feature randomness in order to produce an uncorrelated tree forest whose collective forecast has much better accuracy than any individual tree's prediction. Bagging enhances accuracy of machine learning methods by grouping them together. In this algorithm, during the splitting of nodes it takes only random subset of nodes into an account. When splitting a node, it looks for the best feature from a random group of features rather than the most significant feature. This results into getting better accuracy.

It efficiently deals with the huge datasets. It also solves the issue of overfitting in datasets. It works as follows:

First, it will select random samples from the provided dataset. Next, for every selected sample it will create a decision tree and it will receive a forecasted result from every created decision tree. Then for each result which was predicted, it'll perform voting and through voting it will select the best predicted result.

### 3.3.2 Logistic Regression

Logistic regression is often used a lot of times in machine learning for predicting the likelihood of response attributes when a set of explanatory independent attributes are given. It is used when the target attribute is also known as a dependent variable having categorical values like yes/no or true/false, etc. It is widely used for solving classification problems. It falls under the category of supervised machine learning. It efficiently solves linear and binary classification problems. It is one of the most commonly used and easy to implement algorithms. It's a statistical technique to predict classes which are binary. When the target variable has two possible classes in that case it predicts the likelihood of occurrence of the event. In our dataset the target variable is categorical as it has only two classes-yes/no.

### 3.3.3 Naïve Bayes

It is a probabilistic machine learning algorithm which is mainly used in classification problems. It's based on Bayes theorem. It is simple and easy to build. It deals with huge datasets efficiently. It can

solve complicated classification problems. The existence of a specific feature in a class is assumed to be independent of the presence of any other feature according to naïve Bayes theorem. The formula is:

$$P(S|T) = P(T|S) * P(S) / P(T)$$

Here, T is the event to be predicted, S is the class value for an event. This equation. will find out the class in which the expected feature for classification.

### 3.3.4 Support Vector Machine (SVM)

It is a powerful machine learning algorithm that falls under the category of supervised learning. Many people use SVM to solve both regression and classification problems. The primary role of SVM algorithm is that it separates two classes by creating a line of hyperplanes. Data points which are closest to the hyperplane or points of the data set that, if deleted, would change the position of dividing the hyperplane are known as support vectors. As a result, they might be regarded as essential components of the data set. The margin is the distance between hyperplane and nearest data point from either collection. The goal is to select the hyperplane with the maximum possible margin between it and any point in the training set increasing the likelihood of a new data being properly classified. SVM's main objective is to find a hyperplane in N-dimensional space which will classify all the data points. The dimension of a hyperplane is actually dependent on the quantity of input features. If input has two features in that case the hyperplane will be a line and two-dimensional plane.

### 3.3.5 K Nearest Neighbor (KNN)

KNN is a supervised machine learning algorithm. It assumes similar objects are nearer to one another. When the parameters are continuous in that case KNN is preferred. In this algorithm it classifies objects by predicting their nearest neighbor. It's simple and easy to implement and also has high speed because of which it is preferred over the other algorithms when it comes to solving classification problems. The algorithm classifies whether or not the patient has disease by taking the heart disease dataset as an input. It takes input parameters like age, sex, chol, etc and classify person with heart disease.

### 3.3.6 eXtreme Gradient Boost

It is a class of ensemble machine learning algorithms which is mostly used to solve classification problems. The gradient is mainly useful for reducing loss function which is nothing but actual diff. between original values and predicted values. Gradient boost is a greedy algorithm that can easily overfit the training dataset in a quick time which results in improving the performance of an algorithm. Boosting is nothing but a type of ensemble machine learning model in which new models of decision trees are added to correct the errors made by previous existing models. In our research we used the XGBoost machine learning algorithm.

XGBoost stands for eXtreme Gradient Boost. It's an open-source library which provides specific implementation of a gradient boost technique. It's a more regularized form of the gradient boost. It is much better in performance than the Gradient boost algorithms because it uses more advanced regularization (L1 & L2). It's one of the most popular algorithms in the machine learning field because it delivers the highest performance in terms of accuracy most of the time than any other algorithm. Its execution speed is really fast.

## 4. Implementation

As we already discussed in the methodology section about some of the implementation details. So, the language used in this project is Python programming. We're running python code in anaconda navigator's Jupyter notebook. Jupyter notebook is much faster than Python IDE tools like PyCharm or Visual studio for implementing ML algorithms. The advantage of Jupyter notebook is that while writing code, it's really helpful for Data visualization and plotting some graphs like histogram and heatmap of correlated matrices.

The implementation steps are:

a) Dataset collection.
b) Importing Libraries : Numpy, Pandas, Scikit-learn, Matplotlib and Seaborn libraries were used.
c) Exploratory data analysis : For getting more insights about data.
d) Data cleaning and preprocessing : Checked for null and junk values using isnull() and isna().sum() functions of python.In Preprocessing phase, we did feature engineering on our dataset. As we converted categorical variables into numerical variables using get_dummies() function of Pandas library. Both our datasets contains some categorical variables

e) Feature scaling: In this step, we normalize our data by applying Standardization by using StandardScalar() and fit_transform() functions of scikit-learn library.
f) Model selection : We first separated X's from Y's. X's are features or input variables of our datasets and Y's are dependent or target variables which are crucial for predicting disease. Then using by the importing model_selection function of the sklearn library, we splitted our X's and Y's into train and test split using train_test_split() function of sklearn. We split 80% of our data for training and 20% for testing.
g) Applied ML models and created a confusion matrix of all models.
h) Deployment of the model which gave the best accuracy.

# 5. Analysis of the Results

We used precision, F1-score, recall and accuracy evaluation metrics for evaluating our models. False Positive (FP) is when a model incorrectly predicts a positive outcome. False Negative (FN) is when a model incorrectly predicts the negative outcome. True Positive(TP) is when model correctly predicts a positive outcome. True Negative (TN) is when a model correctly predicts a negative outcome.

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

## 5.1 Results of the First Dataset [UCI heart disease dataset]

Following are the results which we obtained after applying ML models on Dataset 1 which is UCI Heart disease dataset containing 303 records and 14 attributes.

Table 3: Results of ML models on UCI dataset

| ML models | Precision | F1-score | Recall | Training accuracy | Testing accuracy |
|---|---|---|---|---|---|
| Naïve Bayes | 0.88 | 0.85 | 0.82 | 83.06% | 86.89% |
| Logistic Regression | 0.97 | 0.91 | 0.87 | 87.60% | 90.16% |
| SVM | 0.97 | 0.93 | 0.89 | 85.54% | 91.80% |
| KNN | 0.94 | 0.91 | 0.88 | 100% | 90.16% |
| Random Forest | 0.95 | 0.92 | 0.87 | 100% | 90% |
| XGBoost | 0.91 | 0.88 | 0.86 | 100% | 88.52% |

Table 3 shows that KNN gives us the best result with training accuracy of 100% and testing accuracy of 90.16%. SVM actually gave us the best testing accuracy of 91.80% but we got less training accuracy. Also, the other random forest and Logistic regression models performed well.

Table 4: Results of ML models on Kaggle (comprehensive) dataset

| ML models | Precision | F1-score | Recall | Training accuracy | Testing accuracy |
|---|---|---|---|---|---|
| Naïve Bayes | 0.82 | 0.78 | 0.77 | 83.93% | 79.83% |
| Logistic Regression | 0.87 | 0.83 | 0.79 | 85.71% | 84.45% |
| SVM | 0.87 | 0.87 | 0.87 | 100% | 90.76% |
| KNN | 0.90 | 0.88 | 0.86 | 100% | 91.60% |
| Random Forest | 0.95 | 0.93 | 0.91 | 100% | 94.12% |
| XGBoost | 0.95 | 0.92 | 0.90 | 100% | 92.86% |

## 5.2 Results of the Second dataset [Kaggle dataset (combination of 5 popular heart disease dataset)]

Following are the results which we obtained after applying ML models on Dataset 2 which is a Kaggle Heart disease dataset(combination of 5 popular dataset) containing 1190 records and 12 attributes.

For dataset 2. table 4 shows that Random Forest Outperformed all the other algorithms by giving us the best testing accuracy of 94.12% and training 100%. XGBoost also gave us 92.86% of testing accuracy. SVM, KNN algorithms also performed well. The only algorithm which performed poorly in this case was Naïve Bayes in comparison to other algorithms. We applied Hyperparameter tuning which played a crucial role in improving the accuracy of the applied algorithms.

C. Results of the Combined dataset [UCI heart disease dataset + Kaggle dataset (combination of 5 popular heart disease dataset)]

Table 5: Results of ML models on Combined Dataset (Dataset1 + Dataset2)

| ML models | Precision | F1-score | Recall | Training accuracy | Testing accuracy | ML models |
|---|---|---|---|---|---|---|
| Naïve Bayes | 0.75 | 0.75 | 0.76 | 78.31% | 77.93% | Naïve Bayes |
| Logistic Regression | 0.79 | 0.76 | 0.74 | 81.99% | 77.26% | Logistic Regression |
| SVM | 0.86 | 0.83 | 0.80 | 99.50% | 83.61% | SVM |
| KNN | 0.78 | 0.80 | 0.81 | 100% | 82.27% | KNN |
| Random Forest | 0.92 | 0.92 | 0.93 | 100% | 93.31% | Random Forest |
| XGBoost | 0.88 | 0.91 | 0.94 | 100% | 91.64% | XGBoost |

Here, we combined both the datasets which we used in this research (Dataset1 and Dataset2) to find overall accuracy. Dataset 1 (UCI heart disease dataset) has 303 records and 14 attributes and on Dataset 2 which is Kaggle Heart disease dataset (combination of 5 popular dataset) containing 1190 records and 12 attributes. So, at the time of combining both datasets we only used 12 attributes of dataset1 because we combined it with a dataset2 which has 12 attributes. Our combined dataset has total of 1493 records and 12 attributes. Following

are the results which we obtained after applying ML models on Combined dataset:

As shown in table 5, for this combined dataset Random Forest gave us the best testing accuracy of 93.31% and training 100%. XGBoost also gave us 91.64% of testing accuracy. Naïve Bayes and Logistic Regression did not perform well in this case.

## 6. Conclusion and Future Work:

In this research, we used six different machine learning algorithms such as Naïve Bayes, K Nearest Neighbor, Support Vector Machine, Logistic Regression, Extreme Gradient Boost, and Random Forest for prediction of heart disease. We used two datasets which are publicly available, one from the UCI machine learning repository and another from Kaggle. For the UCI heart dataset, we got the highest test accuracy for SVM of 92%. When we applied ML models on the second dataset(Kaggle) we got the highest accuracy of 94.12% by using the Random Forest algorithm, while the lowest accuracy we got for Naïve Bayes of 79.83%. Then we combined both the datasets which we used in our research for which we got highest accuracy of 93.31% using Random Forest. In all cases, Random Forest, KNN, SVM, and XGBoost algorithms performed really well. The only exception is the Naïve Bayes algorithm. Hyperparameter tuning really helped to get improved accuracy as it helped to find the best parameters for the ML model. Also the feature scaling and feature engineering steps also played a crucial role in giving us the best accuracy. For Performance measurement, I used precision, recall, TN, FP, FN, TP rate, evaluation metrics. Based on the results which we got in our research it indicates that applying machine learning algorithms are really important to predict heart disease in an early stage. Our system could assist doctors in diagnosing heart disease. In the future, we may investigate deep learning algorithms to see the results.

*Reference:*

[1] Umair Shafique, Fiaz Majeed, Haseeb Qaiser, Irfan Ul Mustafa on “Data Mining in Healthcare for Heart Diseases”, International Journal of Innovation and Applied Studies ISSN 2028-9324 Vol. 10, Issue 4, Mar. 2015, pp. 1312-1322

[2] V. Sabarinathan on “Diagnosis of Heart Disease Using Decision Tree”, International Journal of Research in Computer Applications & Information Technology Volume 2, Issue 6, November-December 2014, pp. 74-79

[3] Vikas Chaurasia on “Data Mining Approach to Detect Heart Diseases”, International Journal of Advanced Computer Science and Information Technology (IJACSIT) Vol. 2, Issue 4, 2013, Page: 56-66, ISSN: 2296-1739

[4] G. Sarika Sindhu et al. “Analysis and Prediction of Cardiovascular Disease using Machine Learning Classifiers”, International Conference on Advanced Computing & Communication Systems (ICACCS) April 2020.

[5] Malkari Bhargav and J. Raghunath “A Study on Risk Prediction of Cardiovascular Disease Using Machine Learning Algorithms”, International Journal of Emerging Technologies and Innovative Research (www.jstor.org), ISSN:2349-5162, Vol.7, Issue 8, page no.683-688, August 2020

[6] Gayathri Ramamoorthy et al. “Analysis of Heart Disease Prediction using Various Machine Learning Techniques”, International Conference on Artificial Intelligence, Smart Grid and Smart City Applications, (AIS GSC 2019)

[7] Apurb Rajdhan et al. “Heart Disease Prediction using Machine Learning”, International Journal of Engineering Research & Technology (IJERT)ISSN: 2278-0181 Vol. 9 Issue 04, April-2020

[8] Hana H. Alalawi and Manal S. Alsuwat “Detection of Cardiovascular Disease using Machine Learning Classification Models”, International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 10 Issue 07, July-2021

[9] J. Maiga, G. G. Hungilo, and others, “Comparison of Machine Learning Models in Prediction of Cardiovascular Disease Using Health Record Data,” in 2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS), 2019, pp. 45–48

[10] A. Lakshmanarao, Y. Swathi, P. Sri Sai Sundareswarar on 'Machine Learning Techniques For Heart Disease Prediction ' in 2019 International Journal of Scientific & technology Research Vol. 8, Issue 11, November 2019 ISSN 2277-8616

[11] Ashok Kumar Dwivedi, "Analysis of computational intelligence techniques for diabetes mellitus prediction," Neural Comput. Appl., vol. 13, Issue 3, pp. 1–9, 2017

[12] Muhammad Saqlain et al. "Identification of Heart Failure by Using Unstructured Data of Cardiac Patients," 2016 45th Int. Conf. Parallel Process. Work., pp. 426–431, 2016

[13] Hossam Meshref on " Cardiovascular Disease Diagnosis: A Machine Learning Interpretation Approach " in 2019 (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, Issue 12, 2019

[14] Baban.U. Rindhe, Nikita Ahire and others "Heart Disease Prediction Using Machine Learning " in 2021 International Journal of Advanced Research in Science, Communication and Technology (IJARCET) Vol. 5, Issue 1, May 2021

[15] H. Jayasree et al. "Heart Disease Prediction System" in 2019 JASC: Journal of Applied Science and ComputationsVolume Vol. 1, Issue 6, JUNE/2019 ISSN NO: 1076-5131

**Author Contributions:**

Nikhil Bora: Graduate Student doing his thesis. His role was doing the research and implementation of this paper. This paper was part of his master thesis.

Sreedevi Gutta: Advisor of Shubha. Dr. Gutta was the advisor of Nikhil and provided valuable comments every week mainly on sections 3 and 5.

Ahmad Hadaegh: "co-advisor". Dr. Hadaegh also directed Nikhil in his work providing valuable feedback mainly on sections 4 and 5.

## Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)