

## Heart Disease Prediction using Machine Learning

S.Nandhini<sup>1</sup>, Monojit Debnath<sup>2</sup>, Anurag Sharma<sup>3</sup>, Pushkar<sup>4</sup>

<sup>1</sup>Assistant Professor,

M.E, SRM Institute Of Science Of Technology,  
Chennai, Tamil Nadu

<sup>2,3</sup>Department of Computer science and Technology,  
SRM Institute Of Science Of Technology,  
Chennai, Tamil Nadu

<sup>4</sup>Department of Electronics and Communication Engineering,  
SRM Institute Of Science Of Technology,  
Chennai, Tamil Nadu

---

**Abstract:** Over the last decade heart disease is the main reason for death in the world. Almost one person dies of Heart disease about every minute in India alone. In order to lower the number of deaths from heart diseases, there has to be a fast and efficient detection technique. Decision Tree is one of the effective data mining methods till this date. The algorithm used in this project is namely are Decision Tree, Naïve Byes, Support vector machine(SVM), k-nearest neighbours algorithm (KNN), Logistic regression, Random Forests. Heart disease defines several healthcare conditions that are vast in nature which is related to the heart and has many basic causes that affect the entire body. The existing data of heart disease patients from Cleveland database of UCI repository is used to make a test and clearance to the performance of decision tree algorithms. These datasets consist of 303 instances and 76 attributes. This study's goal is to predict the presence of heart disease in patients where this presence is valued from no presence to likely presence. In this project, we proposed a vertical system integration of a sensor node and toolkit of machine learning algorithm for predicting the heart diseases of a person. The dataset for this project is automatically taken from the raw value of the heart pulse sensor and it also used some manually given data. With this project, we have shown that the raw data from the sensor actually increases the accuracy of the model.

**Keywords:** Machine Learning, AI algorithms, Heart Attack, Cardiovascular Disease, Cleveland Dataset, Smart monitoring system

---

### 1. Introduction

One of the most important organs of the human body is a heart. one of the most common cardiac diseases in India is the heart attack. The heart pumps blood through the circulatory system of the body. In all body part the blood, oxygen is circulated by the circulatory system of the body and if the heart does not work properly then the whole human blood system will be collapsed. So if the heart does not function properly then it will lead to a serious health condition, it could even lead to death.

#### 1.2 Types of the Heart Disease:

CardioVascular disease (CVD) or also known as heart disease include blood and heart of the human body. myocardial infarction (as a heart attack) is also a part of the CVD. Another type of Heart Disease is called Coronary Heart Disease(CHD), in this type of disease, a substance called Plaque develop in the coronary arteries. The development of plaque can block the vessel completely through the course of time. The symptoms of the Heart Attack :

1. Chest Pain: The most common sign of a heart attack is chest pain. It mainly happens cause of the blockage of the coronary vessel of the body due to the plaque.
2. Arms pain: The pain normally starts in the chest and move towards the arm mainly left arm.
3. Low in oxygen: Because of the plaque the level of oxygen drops in the body and causes the dizziness and loss of balance.
4. tiredness: this cause for fatigues means simple chores become harder to do.
5. Excessive Sweating: Another common symptom is sweating.
6. Diabetics: In this, the patients have a heart rate of ~ 100 bpm and also occasionally having a heart rate of 130bpm.
7. Bradycardia: In this, the patient will have a slower heartbeat of 60 bpm.
8. Cerebrovascular Disease: The patient will have a high heart rate than normal usually of 200 bpm and higher than this can cause a Heart attack.

9. Hypertension: In this the patient's heart rate normally ranging from 100-200 bpm.

Some other reasons for the occurrence of a heart disease are lifestyle habits like smoking and certain eating habits. An estimated assumption is that more than 17.5 million deaths occur because of cardiovascular disease in the whole world. In India, there is more than 30 million heart disease patient right now. In India, more than 2 lake open heart surgeries are done per year. The patients affected by the heart attack is growing in India is 20% to 30 % every year. The development of the sensor network in the human monitoring system is more applicable from recent years. In this project, we show the use of sensor collected data which automatically generated by the sensor how it can be applied in Machine learning algorithms. Our project is based on the data of the sensor which collects the human heart rate. By using the sensor data and applying it in Machine learning algorithm we will predict the heart disease. The rest of the paper is described in the following manner. Section 2 describes the proposed system. Section 3 describes the Architecture. Section 4 includes experimental result and analysis. Section 5 contains conclusion and future work. Section 6 contains the acknowledgement. Section 7 contains references.

### 2. Proposed system

In this project, we propose a system which can be used for both heart disease monitoring and diagnosis. In this project, the proposed system can notify if an emergency situation occurs. the pulse rate sensor(AMPED) and Bluetooth 4.0 fitted to it and sends the data to the mobile application. Now the sensor data is pushed to the cloud for further analyzing of Heart Rate Variability (HRV). The graph of the heart rate and the prediction of heart disease can be seen through the mobile app. Based on the increase or decrease of the HRV the model will be predicting the chances of the disease to happen and the notification will be sent to the user's phone.

### 3. System Architecture :

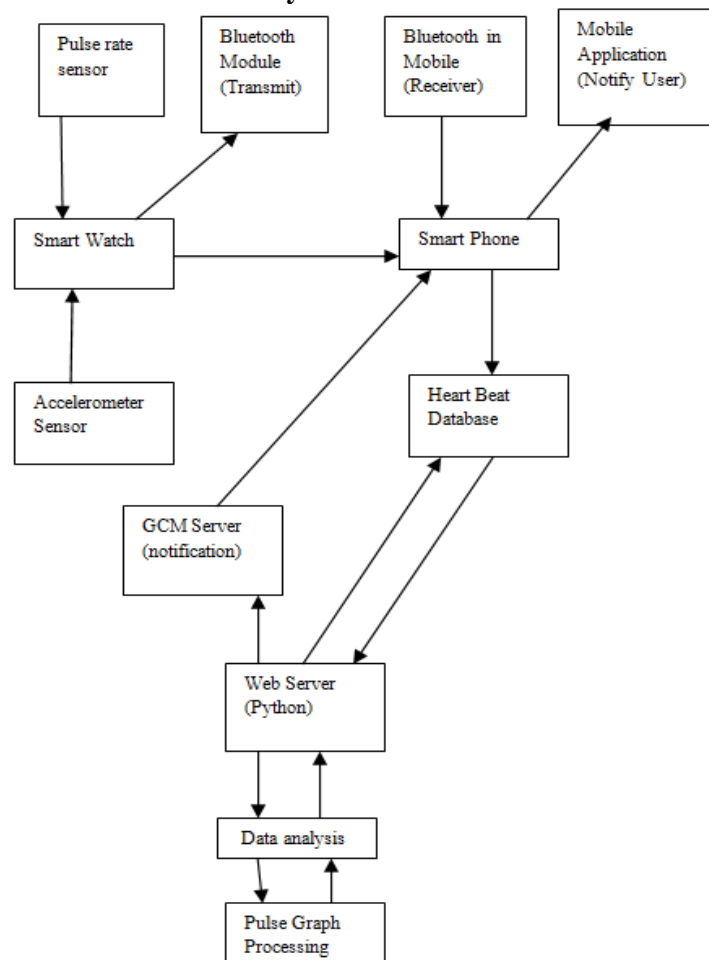


Figure1: system architecture

### 3.1 Assumption

In this project, we assume that mobile is always connected to Bluetooth with mobile device and internet

### 3.2 Hardware Requirement

Pulse sensor (Amped), Bluetooth HC-06 module.

### 3.3 Software Requirements

Arduino suite, Android Studio, MongoDB.

### 3.4 Sensors

This project consists of a pulse sensor AMPED. This sensor combines the of a simple optical Heart rate sensor with amplification and a noise cancellation circuitry for making it fast and easy for a reliable heart pulse readings. This sensor was chosen for this project because of its low cost and reliability, also the sensor is energy efficient with a power consumption of only ~4mA at 5V with voltage ranging from 3V to 5V.

### 3.5 Gateway to WANs

WAN is responsible for connecting the sensing component to the infrastructural WANs, collected data from the sensor should be transmitted through a gateway, normally gateways are a mobile phone, personal computer, a remote site such as a hospital. In this project we carried out the connection by firstly connecting the sensor to an Arduino microcontroller, then the sensing data is transmitted to a mobile phone. In this project, we used an HC-06 Bluetooth module which is integrated with a microcontroller. This Bluetooth module uses UART(Universal Asynchronous Receiver/Transmitter ) protocol. Sending data and receiving data is more reliable through UART. When an emergency situation occurs an emergency alert will be sent to the nearest hospital through the mobile internet. The nearest hospital is detected by the GSM system of the mobile. For relaying the message to the hospital it will need the internet with board area communication protocols.

### 3.6 The End-user Healthcare Centre

This part of the system is where all the data that were collected is analyzed and the action gets triggered. Diagnosis and Monitoring are two main components for our project. Figure 2 shows the monitoring result in our application.

#### 3.6.1 The Diagnosis Component

The diagnosis component is able to successfully predict the heart disease based on the given input by doctors and patients. This smart component is created based on the Cleveland clinical data. Input data are shown in Figure 2.



Figure 2: input and output display of the app

Disease dataset which is taken from the Cleveland heart disease repository. Based on the dataset and by the use of machine learning algorithms we have attained the classification and a system is created that can predict whether a heart disease is present or absent for a new user. In this work, we have used Support Vector Machines(SVM), Decision Trees, Random Forest, Logistic Regression, Naive Bayes, Nearest Neighbour(KNN).

**Naive Bayes Classifier:** It is a classifier technique based on "Bayes theorem", it assumes a particular class which has a particular feature is unrelated to other features in class. We used it in the model because it is easy to

make and useful for large dataset. It provides data structures such as Network Structure, Conditional probability distribution and others. Figure 3 shows the naive Bayes code.

```
#Gaussian Naive Bayes
model = train_model(X_train, y_train, X_test, y_test, GaussianNB)

Train accuracy: 65.36%
Test accuracy: 66.81%
```

Figure 3 Naive Bayes Train and Test accuracy

**Support Vector Machines(SVM):** A Support Vector Mechanism (SVM) is a discriminative classifier formally defined by a separating hyperplane. Simply put the algorithm outputs an optimal hyper plane which categorises new examples. In 2-D space, this hyper plane is a line dividing a plane into two parts wherein each class lay on either side. The points which positions are in the separating Hyperplane is called Support Vectors. The distance between the Canonical and Separating hyperplane is called Margin. We have implemented a variant of SVM which is sequential minimal optimization. The minimal sequential optimization breaks the problems in subproblems and then solve it analytically.

**Logistic Regression:** Logistic Regression is a method which analyses a dataset which has a one or more independent variable and gives an outcome. The goal of the Logistic Regression is to predict the best relationship between the dependent and independent variables. Figure 3 shows the Logistic Regression of the model and the accuracy of the test and train model.

```
# Logistic Regression
model = train_model(X_train, y_train, X_test, y_test, LogisticRegression)

Train accuracy: 85.85%
Test accuracy: 85.71%
```

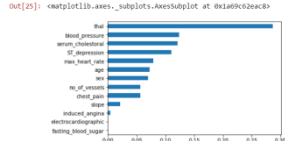
Figure 4 Logistic Regression

**Decision Tree:** Decision tree is used for making a tree like structures for regression or classification models. A decision tree creates a smaller and smaller subset of a problem while an associated decision tree is developed incrementally. Two or more branches and leaf can seem in a decision tree which represents classification. Both categorical and numerical value can be handled by a decision tree. The algorithm Decision tree can learn to predict the value of a target variable by learning simple decision rules taken from the dataset. From the result of our decision tree, we can easily understand how much importance a particular feature has. In figure 5 we can see the feature 'Thal' is turned out to be a very important feature of our model.

```
In [25]: # Decision Tree
model = train_model(X_train, y_train, X_test, y_test, DecisionTreeClassifier, random_state=2000)
# plot feature importances
plt.bar(sorted(model.feature_importances_,X.columns).sort_values(ascending=True).plot.barh())

Train accuracy: 100.00%
Test accuracy: 75.82%
```

out[25]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1a69c02eac8>



Feature	Importance (approx.)
Thal	0.28
blood_pressure	0.18
serum_cholesterol	0.15
SG_cholesterol	0.12
max_heart_rate	0.10
age	0.08
sex	0.05
no_of_ventricles	0.04
chest_pain	0.03
diabetes	0.02
electrocardiographic	0.01
heart_rate_max	0.01

Figure 5 Decision Tree prediction of the feature

Here the decision tree learns the train set model perfectly and overfitting the data. That's why it will give a poor prediction. Other values of 'max\_depth' parameter need to be tried out, it is shown in Figure 6.

```
In [49]: # Seek optimal 'max_depth' parameter
for i in range(1,8):
    print("max_depth = "+str(i))
    train_model(X_train, y_train, X_test, y_test, DecisionTreeClassifier, max_depth=i, random_state=2000)

max_depth = 1
Train accuracy: 76.80%
Test accuracy: 74.73%
max_depth = 2
Train accuracy: 78.30%
Test accuracy: 72.53%
max_depth = 3
Train accuracy: 87.74%
Test accuracy: 76.92%
max_depth = 4
Train accuracy: 91.96%
Test accuracy: 78.82%
max_depth = 5
Train accuracy: 94.81%
Test accuracy: 78.82%
max_depth = 6
Train accuracy: 87.17%
Test accuracy: 79.12%
max_depth = 7
Train accuracy: 87.64%
Test accuracy: 75.82%

With max_depth set as 6, the score went to almost 80%. By now, KNN outperforms Decision Tree.
```

Figure 6 'max\_depth' parameter With 'max\_depth' six the score went to almost 80% of the decision tree.

**K- Nearest Neighbour (KNN):** KNN is a supervised classification algorithm (it takes a bunch of labeled points and uses them how to label another point).To label a new point it looks at the new point nearest to it and votes for it and whichever label is the most voted that label is given to the new point. Below in figure 7we can see KNN model

```
# KNN
model = train_model(X_train, y_train, X_test, y_test, KNeighborsClassifier)

Train accuracy: 88.21%
Test accuracy: 86.81%
```

Figure 7 KNN train and test accuracy

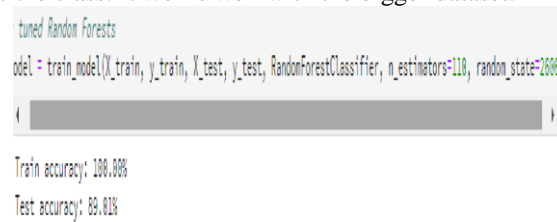
```
# Seek optimal 'n_neighbors' parameter
for i in range(1,10):
    print("n_neighbors = "+str(i))
    train_model(X_train, y_train, X_test, y_test, KNeighborsClassifier, n_neighbors=i)

n_neighbors = 1
Train accuracy: 88.88%
Test accuracy: 74.73%
n_neighbors = 2
Train accuracy: 87.74%
Test accuracy: 79.12%
n_neighbors = 3
Train accuracy: 98.87%
Test accuracy: 83.52%
n_neighbors = 4
Train accuracy: 87.74%
Test accuracy: 84.82%
n_neighbors = 5
Train accuracy: 88.21%
Test accuracy: 86.81%
n_neighbors = 6
Train accuracy: 85.58%
Test accuracy: 86.81%
n_neighbors = 7
Train accuracy: 87.26%
Test accuracy: 86.81%
n_neighbors = 8
Train accuracy: 85.38%
Test accuracy: 85.71%
n_neighbors = 9
Train accuracy: 88.32%
Test accuracy: 85.71%
```

Figure 8 'n\_Parameter'

Despite its simplicity, the result is very good so we put different values for n.

**Random forest:** Random Forest algorithm does not overfit the set like 'Decision Tree'. Random Decision Tree first considers many decision trees before giving an output. Random forest algorithm uses a voting system for classification where it decides the class. It works well with the bigger dataset.



```
tuned Random Forests
model = train_model(X_train, y_train, X_test, y_test, RandomForestClassifier, n_estimators=110, random_state=2000)

Train accuracy: 100.00%
Test accuracy: 89.01%
```

Figure 9 Random Forest

### 3.5.2. The Monitoring System

The monitoring system is a component where the reasoning performs with the help of algorithmic computation for continuously checking the patient's heart rate and save the details to a log file. The saved data can be review by the user and the doctor when needed. Once for if an emergency situation occurs an email is generated with details of the patients and send to the nearest hospital with the exact location of the patient and this is done by the help of the GPS system. The location of the user will be updated every one minute. The emergency email contains the user's age, sex, name, mobile phone num. Most people have a resting heart rate of 60 to 100 beats per stated by American Science Institute. When someone playing sports the heart rate can be increased up to 150 bpm. We have implemented this values in our model and the system will be raising an alarm whenever the before mentioned values will be crossed. In this model, there are two parts namely rest mode and play mode. This mode will be detected by the accelerometer attached to the watch. If the model detects any emergency situation staying for a widow of five minutes then that generates an email and an SMS and will be sent to the nearest hospital and relative of the patient.

## 4. Experimental Result And Analysis

The experiment is conducted for diagnosis and monitoring features of the model. The experiment was conducted with Samsung J8 model. It has octa-core processor and a ram of 4 GB and a storage if 16 GB.

### 4.1 Diagnosis Experiment

In this project, we are making an intelligent classifier that is able to detect the new patient and assign it to one of the two classes no presence or the presence of the disease.

### 4.2 The Dataset

For this project we have used The Cleveland heart dataset from the UCI Machine Learning Repository as it is widely used by the Pattern design community. The dataset consists of 303 individual clinical reports in which 164 did not have any disease. In this dataset there is a total of 97 female patients in which 25 people are the affirmative case, also there are 206 male patients in which 114 are diagnosed with the disease. There is 6 missing value in this dataset and all numeric values are recognized as numeric. It is shown in Figure10.

```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 18 columns):
age          303 non-null float64
sex         303 non-null float64
chest_pain  303 non-null float64
blood_pressure  303 non-null float64
serum_cholsterol  303 non-null float64
fasting_blood_sugar  303 non-null float64
electrocardiogramic  303 non-null float64
max_heart_rate  303 non-null float64
diastolic_blood_pressure  303 non-null float64
ST_depression  303 non-null float64
slope       303 non-null float64
no_of_vessels  299 non-null float64
trest      303 non-null float64
diagnosis   303 non-null int64
dtypes: float64(13), int64(1)
memory usage: 33.2 KB
```

Figure 10 'missing values' of the dataset

Thus we have 13 features that are shown below

- Age
- sex
- Chest Pain Type
- Resting Blood Pressure
- Serum Cholesterol in mg/dl
- Fasting Blood Sugar
- Resting electrocardiographic result
- Maximum heart rate achieved
- Exercised-induced angina
- Old peak, ST depression induced by exercise relative to rest
- Number of major vessels coloured by fluoroscopy
- Thal:3= Normal , 6=fixed defect , 7= reversible defect

#### 4.2.1. Hold Out Test

Sometimes referred to as testing data, the holdout data provide the final estimation of the machine learning model performance after it has been trained. In this model, we divided the 303 patients into two parts. In the hold, an experiment, we almost used 2/4 data for training and build the classification model.

```
def train_model(X_train, y_train, X_test, y_test, classifier, **kwargs):
    """
    Fit the chosen model and print out the score.
    """
    # instantiate model
    model = classifier(**kwargs)
    # train model
    model.fit(X_train, y_train)
    # check accuracy and print out the results
    fit_accuracy = model.score(X_train, y_train)
    test_accuracy = model.score(X_test, y_test)
    print(f"Train accuracy: {fit_accuracy:0.2%}")
    print(f"Test accuracy: {test_accuracy:0.2%}")
    return model
```

Figure 11 Test Set

Figure 11 shows the test model. After this, the original data is compared to the predicted data and then the accuracy is calculated.

	accuracy
<b>KNN</b>	0.868132
<b>Decision Trees</b>	0.758242
<b>Logistic Regression</b>	0.857143
<b>Naive Bayes</b>	0.868132
<b>SVM</b>	0.879121
<b>Random Forests</b>	0.890110

Figure 12 Accuracy of the classifiers

As we can see here in Figure 12 the most accuracy is having the classifier, Random Forest. The positive rate of the Random Forest is 89% and the false positive rate is 1.1 %.

#### 4.2.2. Monitoring Experiment

In this project to test the model, we asked 20 healthy individuals here where 5 of them were told to play sports while carrying the model. Cause all the users were healthy that's why no threshold divination occurred so no alarm was raised. The first two row in Figure 13 is showing 100% accuracy in the detection result of the healthy person. In this, the BPM detected values are compatible with real values. For further testing our model we used this on 20 unhealthy individuals in both the rest and lay mode. This time the threshold has deviated and the alarm raised and SMS sent to the nearest hospital moreover GPS location was detected accurately in all the cases. This also an accuracy of 100% was gain.

In any healthcare application, one main material is privacy. Here we have implemented an end to end privacy mechanism that only allows the patient or the doctor to see the details of the patient on the approval of the patient.

## 5. Conclusion and Future work

In this paper, we have presented a system which is suitable for real-time heart diseases prediction and can be used by the users who have coronary disease. Different from many other systems it is able to both monitor and prediction. The diagnosis system of the system is able to predict the heart disease by using ML algorithms and the prediction results are based on the heart disease dataset instance. On the other hand, the system is very inexpensive, we used amped pulse sensor and send the data to mobile via Arduino suite microcontroller. For checking the variances and raise the alarm if the user's heart rate rise than the normal rate of the heart. To prove the effectiveness of the system we have carried out experiments for both monitoring and diagnosis system . we ran experiments with some popular algorithms like KNN, Decision Tree, Random Forest, Naive Bayes, SVM, Logistic Regression. The experiment was carried out with the holdout test and the accuracy of the proposed system was 89% achieved with the Random forest.

For the monitoring system, we have carried out two experiments. On the first experiment we have experimented with 20 individuals healthy persons and on the second one, we experimented with 20 individuals who have a cardinal disease. In both cases, the accuracy of the monitoring system was 100%. In future, we are planning to use PPG (Photoplethysmography) system and omit the use of a sensor.

## 6. Acknowledgment

We sincerely thank the staff member of SRM Institute of Science and Technology for supporting our project immensely.

### Authors Biography

**First Author-** S.Nandhani, M.E, Assistant professor, SRM Institute Of Science Of Technology, Chennai, Tamil Nadu

**Second Author-** Monojit Debnath, Department of Computer science and Technology, SRM Institute Of Science Of Technology, Chennai, Tamil Nadu

**Third Author-** Anurag Sharma, Department of Computer science and Technology, SRM Institute Of Science Of Technology, Chennai, Tamil Nadu

**Fourth Author-**Pushkar, Department of Electronics and Communication Engineering, SRM Institute Of Science Of Technology, Chennai, Tamil Nadu

**Correspondence Author-** Monojit Debnath

## 7. References

- [1]. [https://www.researchgate.net/publication/319393368\\_Heart\\_Disease\\_Diagnosis\\_and\\_Prediction\\_Using\\_Machine\\_Learning\\_and\\_Data\\_Mining\\_Techniques\\_A\\_Review](https://www.researchgate.net/publication/319393368_Heart_Disease_Diagnosis_and_Prediction_Using_Machine_Learning_and_Data_Mining_Techniques_A_Review)
- [2]. Predicting and Diagnosing of Heart Disease Using Machine Learning Algorithms Sanjay Kumar Sen Asst. Professor, Computer Science & Engg. Orissa Engineering College, Bhubaneswar, Odisha – India
- [3]. Heart disease prediction using machine learning techniques: a survey V.V. Ramalingam\*, Ayantan Dandapath, M Karthik Raja
- [4]. Effective Diagnosis and Monitoring of Heart Disease Ahmed Fawzi Otoom1 , Emad E. Abdallah2 , Yousef Kilani3 , Ahmed Kefaye4 and Mohammad Ashour5 1,2,3,4,5 Faculty of Prince Al-Hussein Bin Abdullah II for Information Technology The Hashemite University, Zarqa, Jordan { 1 bottom, 2 emad,3 ymkilani }@hu.edu.jo, 4 a.kefaye@alpha-hub.com, [5m.ashour@teleogx.com](mailto:5m.ashour@teleogx.com) [https://pdfs.semanticscholar.org/f4ec/b47e080001d8ea08bab686acdb5a741a7159.pdf]
- [5]. Disease Prediction Using Heart rate Variability Analysis - IoT Dharmik Jampala 1, Venkat Naidu Mittapalle2, Sitanshu Nandan3 1, 2, 3 School of Information Technology and Engineering, VIT University, Vellore