Comparison of various machine learning approaches uses in heart ailments prediction

Gunjan Gupta1, Adarsh U2, N V Subba Reddy3, B Ashwath Rao4

1MTech Computer Science & Engineering, MIT, Manipal 576104, India
2MTech Computer Science & Engineering, MIT, Manipal 576104, India
3Professor, Department of Computer Science & Engineering, MIT, Manipal 576104, India
4Assistant Professor(Selection Grade), Department of Computer Science & Engineering, MIT, Manipal, 576104, India

1guptagunjan999@gmail.com, 2adarsh94@gmail.com, 3nvs.reddy@manipal.edu, 4ashwath.rao.b@gmail.com

Abstract. Heart disease has been the leading cause of a huge number of deaths in recent years. As a result, an accurate and feasible system is required to diagnose this disease early to provide better treatment. Advances in machine learning have the potential to enhance healthcare access. Given the importance of a crucial organ like the heart, medical professionals and physicians have made it a priority to forecast heart failure-related events in clinical practice, nevertheless, forecasting heart failure-related events in clinical practice has generally failed to achieve high accuracy. The objective here is to demonstrate how machine learning may be used to solve the problem. By analyzing hundreds of healthcare data and other semantics, machine learning algorithms can analyze related cases with diseases and health conditions. Here a demonstration of how to load the data, generate predictions through different models from patient data is shown. The metrics are then compared for a better understanding of their function and what impact can be inferred from them.

1. Introduction
The heart of the problem or problem of the heart here is mainly the conditions affecting the heart, there are many types, but most are preventable, which brings us to the foray of the discussion that prevention is better than cure. Heart ailments have overtaken cancer as the leading cause of death [1]. A great many individuals pass on every year because of coronary illness. Heart ailments have also become the leading cause of death in India. A heart ailment is a condition that impairs the function of the heart.

The first step of prevention is detection; our paper refers to the observation method as opposed to the physical measurements such as ECG monitoring etc. Using previous data from many patients, to forecast cardiac disorder machine learning approaches are utilized, such as Decision Tree, K-Nearest Neighbor (KNN), Artificial Neural Network (ANN), Naïve Bayes, and Random Forest. We would analyze the various algorithms to find out which is the most effective method to give an accurate outcome, due to digital technologies are rapidly growing, health care centers store huge amounts of
data in their database that is very private and can be accessed by elite database operators while they are difficult to understand.

Data gathering, processing, and machine learning play crucial roles in the prediction of such information in medical platforms. Prognostication of the presence of cardiac disease in the human body is the ultimate target of this model. We are using Machine Learning to make predictions on whether a person would be suffering from heart disease. Facets like medical, manufacturing, chemical, etc. that measure already taking advantage of data foraging. The experience concludes that in-depth call support needs new technology. This novel technology ought to change the innovation of trends and prognostic patterns in the knowledge of hypotheses and the generation of insight-provoking visualizations. The thought of information mining helps the customers to extract helpful insights from an ocean of data.

2. Literature Survey
Artificial intelligence and machine learning algorithms have acquired a lot of traction because of their boosted accuracy and efficiency in producing predictions [2]. Several studies have been done in past few years over different datasets to evaluate better classification accuracy.

In the model presented by Pahwa et al. [3] principal cause of studies is to categorize the statistics into positive and negative results for coronary heart disease. To optimize the category problem, a hybrid method of function choosing is applied, with combined feature selection and gain-ratio results being deployed to generate a part of capabilities and remove useless functions. To categorize them into the presence or absence of sickness, naive Bayes and random woodland are used on a subset of capabilities. This model has a precision of 84.16% with the 10 most significant attributes.

When the value of k is 9, KNN has an accuracy of 83.16 percent, and the decision tree has meager performance with an exactness of about 78 percent. However, when the decision tree is employed with boosting approach, it has an exactness of about 82 percent [4].

Tahira Mahboob et al. [5] have applied a combination of ANN, SVM, and KNN to yield a correctness of 94.12%. Saba Bashir et al. [6] showed the majority model by which is made up of Naïve Bayes along with a decision tree, which gave an efficacy of 82%.

For forecasting, Ashraf et al. [7] individual learning methods were used as well as ensemble techniques such as Bayes Net, KNN, J48, multilayer perceptron, random forest, Naïve Bayes, and random tree. J48 was the most accurate, with a score of 70.77 percent. They then used cutting-edge approaches, with KERAS achieving an accuracy rate of 80%.

For this classification job, David et al. [8] proposed using the Naive Bayes algorithm, Decision Tree, and Random Forest method as classifiers. They used the SlatLog dataset to analyze the performance of the classifiers and measured the results using various cross-validations. They also utilized evaluation methods based on percentage splits. They discovered that the Random Forest Classifier has the best classification accuracy, 77 percent.

Kahramanli et al. [9] created a neural network that along with neural networks like an artificial network (ANN) and a fuzzy network (FNN) to achieve an efficacy of about 87 percent.

The Intelligent Heart Disease Prediction System (IHDPS) is described in [10]. The system makes use of data extracting techniques like DT, Naïve Bayes, and Neural Networks (NN). According to the results of the authors' testing, the NB model had the optimum implementation in terms of immaculate
predictions (86.12 percent). The penultimate model was NN with correct prediction in 86.12% and the third one with about 80% of correct predictions.

Four algorithms were used in [11] to prepare the dataset and investigate it. The precision for Decision Tree and Random Forest methods was 99.83 percent, while SVM and KNN methods were 85.32 percent and 84.49 percent, respectively.

In our present work, our motto has been to differentiate various techniques, and their use in combinations is observed. This is done to be in a position to select the most optimal forecasting method.

3. Methodology

The suggested system's goal is to employ machine learning techniques to predict whether a person is experiencing a cardiac condition. The figure below depicts the system's architecture. Among the six stages are Collection of data, pre-processing of data, feature selection, splitting of data, training the model, and evaluation.

**Figure 1.** Proposed system structure for heart disease prediction

### 3.1. Dataset collection

The Dataset is collected from Kaggle [12], it offers information about heart disease patients. The dataset contains 303 records, 13 features, and 1 target column. The model is implemented in Jupyter notebook using Python as the programming language. Libraries like Pandas, NumPy, Matplotlib are used for further data pre-processing.

| Attribute | Description                  | Variable Type | Type  |
|-----------|------------------------------|---------------|-------|
| Age       | Age of the person            | continuous    | integer |
| Sex       | Gender Analysed              | categorical   | integer |
| Cp        | Chest Pain Type              | categorical   | integer |
|           | 4 Categories                 |               |       |
| trestbps  | Blood pressure at rest       | continuous    | integer |
| chol      | Cholesterol in mg            | continuous    | integer |
| fbs       | Sugar on an empty stomach    | categorical   | integer |
3.2. Data pre-processing

Before any analysis, we just look at the dataset. So, we used the info () and describe () methods to obtain the details of null and missing values.

To cope with null values in the dataset, we employ the KNN Imputation technique. KNN Imputer uses the K-Nearest Neighbour method to replace missing values in the dataset. K neighbors are chosen based on distance measure (in this implementation Euclidean Distance Matrix is used) and their average is used as an imputation estimate.

Our studied dataset comprises categorical variables that must be addressed before applying any machine learning technique. To work with categorical variables, we must divide each category into dummy columns, the get_dummies () method employed to split each category into dummy columns. This dataset has been scaled.

3.3. Feature selection

Irrelevant characteristics detract from the model's performance. As a result, to improve the accuracy of the model, we determined the dataset's most relevant features. Use the Correlation matrix with Heatmap to discover strongly related features. Cp, thalach, exang, oldpeak, ca, and thal were chosen to train the model since they are significantly linked to the target variable.

3.4. Data splitting

The data set is parted in the proportion of 4:1 where the former is training data and the latter is test data.

3.5. Machine learning model training

In this study, we employed machine learning techniques such as Decision Tree, K Nearest Neighbour, Random Forest, Naïve Bayes, and Artificial Neural Network. Furthermore, by varying their different parameters, two ensemble learning strategies, namely Bagging and Boosting, are used to train the model.

3.5.1. Decision Tree Classifier (DT)

The dependent attribute’s value is determined by the values of the independent features in a decision tree [13]. A Decision Tree is a pictorially measured depiction of all possible answers to a Decision based on specified tests. It includes internal nodes, branches, and a terminal node. All internal nodes have a feature "condition," and branches have the conclusion, and the labels of class are applied for each leaf node. This is used for both classification and regression [14].

3.5.2. Naïve Bayes Classifier (NB)

The fundamental assumption and crucial factor for predicting with this classifier are that the attributes of the data set are independent. For a given categorical variable, the main assumption of
this classifier is that all features are linearly independent of each other, and this point is used for prediction. The Gaussian Nave Bayes (GNB) classifier is well-known for pattern recognition and prediction. This model works by collecting each data point and assigning it to the class that is close to it. The GNB considers the comparing class variance in addition to the Euclidean distance from the class means [15].

3.5.3. Artificial Neural Network (ANN)
Layered neural networks are one of the common types of neural networks. Layers are built up of interconnected "nodes" that each has an “activation function”. Patterns are supplied through the ‘input layer,’ to network, which connects with ‘hidden layers,’ which do real processing using a weighted connection mechanism. The answer is then produced from the hidden layers, which are linked to an ‘output layer.’ Model here formulated on forward propagation. ‘ReLU’ activation function is used in input and hidden layer along with ‘Sigmoid’ activation function used in the output layer and optimizer used is ‘Adam’.

3.5.4. K-Nearest Neighbors Classifier (KNN)
The KNN classifier searches for the classes of a given point's K nearest neighbors and gives a class to that data point based on the majority class. The number of neighbors, on the other hand, can be modified. The class of a new instance will be predicted using distance metrics. A trivial Euclidean distance metric is deployed in KNN algorithms in this case.

For finding the optimal value of K in the KNN Elbow method is used. The Elbow method plots the cost function produced by different facets of K. However, the improvement in error rate will decline as k increments. The value of K at which the improvement in distorted values becomes low and the maximum is referred to as the elbow, and it is at this value that one should cease exploring into data anymore.

3.5.5. Random Forest Classifier (RF)
Random Forest works by building a different number of Decision Trees and then boosting all the Decision Trees' output to arrive at a single outcome [3]. It generates a forest of trees, each of which is made up of a random selection of features from the overall set. Random Forest may be used to analyze both categorical and data.

Ensemble approaches are a powerful tool for classification that combines a weak and a strong classifier to boost the effectiveness of the weak learner.

3.5.6. Bagging
Bagging divides data into subgroups based on a training sample chosen at random with replacement. Their decision trees are trained with each group of data. As a result, we have a collection of various models. The average of these predictions from various trees is made use of, which is more reliable than a single decision tree classifier.

3.5.7. Boosting
Learners in Boosting are taught systematically, beginning with early learners fitting simple models to information and then examining data for flaws. Trees sequentially are fit with the aim of improving accuracy over the previous tree at each step. When a hypothesis incorrectly classifies an input, its weight is increased such that the next hypothesis is highly likely to correctly perform its function.
3.6. Model Validation and Evaluation
The models are analyzed by evaluating precision, recall, f-measure, and accuracy scores. Their performance on the test data is anticipated through a confusion matrix. AUC Score and ROC curve are used to view the result of performance. It tallies the rate of true positivity over the false-positive rate.
AUC is between 0 and 1. Higher the AUC score, the better the classification model.
Accuracy in terms of true and false binary representation is:

\[
\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total number of samples}}
\]

Recall can be calculated through obtained true and false positives as follows:

\[
\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}
\]

Precision is identified as follows:

\[
\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}
\]

The F-measure is calculated through precision and recall as follows:

\[
F - \text{Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

4. Results
4.1. Decision Tree Classifier (DT)
The accuracy score obtained for this model is 81.97%. The classification report includes recall, f-measure, precision is shown below in figure 2. The desired output for the classifier is displayed in figure 3 in the form of a confusion matrix. The AUC value obtained is 0.82, with the corresponding ROC graph displayed below in figure 4.

![Figure 2. Classification report - Decision Tree Classifier](image)

![Figure 3. Confusion matrix - Decision Tree](image)

![Figure 4. ROC curve - Decision Tree](image)

4.2. Naïve Bayes Classifier (NB)
The accuracy score obtained for the model is 85.25%. Classification report includes recall, f-measure, precision is shown below in figure 5. The desired output for the classifier is displayed in figure 6 in the
form of a confusion matrix. The AUC value obtained is 0.85, with the corresponding ROC graph displayed below in figure 7.

|     | precision | recall | f1-score | support |
|-----|-----------|--------|----------|---------|
| 0.0 | 0.65      | 0.81   | 0.83     | 27      |
| 1.0 | 0.86      | 0.88   | 0.87     | 34      |
| avg / total | 0.85 | 0.85 | 0.85 | 61 |

**Figure 5.** Classification report - naïve bayes Classifier

![Confusion matrix - Naïve Bayes](image)

![ROC curve - Naïve Bayes](image)

4.3. *K*-Nearest Neighbour Classifier (KNN)

The accuracy score obtained for the model is 88.52%. Classification report includes recall, f-measure, precision is shown below in figure 8. The desired output for the classifier is displayed in figure 9 in the form of a confusion matrix. The AUC value obtained is 0.89, with the corresponding ROC graph displayed below in figure 10.

|     | precision | recall | f1-score | support |
|-----|-----------|--------|----------|---------|
| 0.0 | 0.83      | 0.93   | 0.88     | 27      |
| 1.0 | 0.94      | 0.85   | 0.89     | 34      |
| avg / total | 0.89 | 0.89 | 0.89 | 61 |

**Figure 8.** Classification report - KNN Classifier

![Confusion matrix - KNN](image)

![ROC curve - KNN](image)

4.4. Random Forest Classifier (RF)

The accuracy score obtained for the model is 86.89%. Classification report includes recall, f-measure, precision is shown below in figure 11. The desired output for the classifier is displayed in figure 12 in the form of a confusion matrix. The AUC value obtained is 0.87, with the corresponding ROC graph displayed below in figure 13.

|     | precision | recall | f1-score | support |
|-----|-----------|--------|----------|---------|
| 0.0 | 0.65      | 0.85   | 0.85     | 27      |
| 1.0 | 0.88      | 0.88   | 0.88     | 34      |
| avg / total | 0.87 | 0.87 | 0.87 | 61 |

**Figure 11.** Classification report – Random forest Classifier

![Confusion matrix - Random Forest](image)

![ROC curve - Random Forest](image)
4.5. Artificial Neural Network (ANN)
The accuracy score obtained for the model is 85.25%. Classification report includes recall, f-measure, precision is shown below in figure 14. The desired output for the classifier is displayed in figure 15 in the form of a confusion matrix. The AUC value obtained is 0.86, with the corresponding ROC graph displayed below in figure 16.

| precision | recall | f1-score | support |
|-----------|--------|----------|---------|
| 0.0       | 0.80   | 0.89     | 0.84    | 27      |
| 1.0       | 0.90   | 0.82     | 0.86    | 34      |
| avg / total | 0.86   | 0.85     | 0.85    | 61      |

**Figure 14.** Classification report – ANN Classifier

**Figure 12.** Confusion matrix - random forest

**Figure 13.** ROC curve - Random Forest

**Figure 15.** Confusion matrix – ANN

**Figure 16.** ROC curve – ANN

**Figure 17.** Constructed Neural Network
4.6. Bagging Classifier
The accuracy score obtained for the model is 83.61%. Classification report includes recall, f-measure, precision is shown below in figure 18. The desired output for the classifier is displayed in figure 19 in the form of a confusion matrix. The AUC value obtained is 0.83, with the corresponding ROC graph displayed below in figure 20.

| precision | recall | f1-score | support |
|-----------|--------|----------|----------|
| 0.0       | 0.81   | 0.81     | 27       |
| 1.0       | 0.85   | 0.85     | 34       |

avg / total 0.84 0.84 0.84 61

Figure 18. Classification report – Bagging Classifier

Figure 19. Confusion matrix – Bagging

Figure 20. ROC curve - Bagging

4.7. Ada-Boosting Classifier
The accuracy score obtained for the model is 85.25%. Classification report includes recall, f-measure, precision is shown below in figure 21. The desired output for the classifier is displayed in figure 22 in the form of a confusion matrix. The AUC value obtained is 0.84, with the corresponding ROC graph displayed below in figure 23.

| precision | recall | f1-score | support |
|-----------|--------|----------|----------|
| 0.0       | 0.88   | 0.78     | 27       |
| 1.0       | 0.84   | 0.91     | 34       |

avg / total 0.85 0.85 0.85 61

Figure 21. Classification report – Ada-boosting Classifier

Figure 22. Confusion matrix – Ada boosting

Figure 23. ROC curve – Ada boosting

4.8. Gradient Boosting Classifier
The accuracy score obtained for the model is 86.89%. Classification report includes recall, f-measure, precision is shown below in figure 24. The desired output for the classifier is displayed in figure 25 in the form of a confusion matrix. The AUC value obtained is 0.87, with the corresponding ROC graph displayed below in figure 26.
Figure 24. Classification report – gradient boosting Classifier

Figure 25. Confusion matrix – Gradient Boosting

Table 2. Confusion matrix values of algorithm’s

| Algorithm           | True Positive | False Positive | True Negative | False Negative |
|---------------------|---------------|----------------|---------------|----------------|
| Decision Tree       | 28            | 6              | 22            | 5              |
| Naïve Bayes         | 30            | 4              | 22            | 5              |
| KNN                 | 29            | 5              | 25            | 2              |
| Random Forest       | 30            | 4              | 23            | 4              |
| ANN                 | 28            | 6              | 24            | 3              |
| Bagging             | 29            | 5              | 22            | 5              |
| Ada-Boosting        | 31            | 3              | 21            | 6              |
| Gradient-Boosting   | 28            | 6              | 25            | 2              |

Table 3. Performance of algorithm’s

| Algorithm          | Precision | Recall | F-Measure | Accuracy    |
|--------------------|-----------|--------|-----------|-------------|
| Decision Tree      | 0.82      | 0.82   | 0.82      | 81.97%      |
| Naïve Bayes        | 0.85      | 0.85   | 0.85      | 85.25%      |
| KNN                | 0.89      | 0.89   | 0.89      | 88.52%      |
| Random Forest      | 0.87      | 0.87   | 0.87      | 86.89%      |
| ANN                | 0.86      | 0.85   | 0.85      | 85.25%      |
| Bagging            | 0.84      | 0.84   | 0.84      | 83.61%      |
| Ada-Boosting       | 0.85      | 0.85   | 0.85      | 85.25%      |
| Gradient-Boosting  | 0.88      | 0.87   | 0.87      | 86.89%      |
4.9. Accuracy Comparison of machine learning techniques

![Accuracy Comparison Diagram]

Figure 27. A diagrammatic comparison of the predicting accuracies (%)

4.10. AUC Score-ROC Curve Comparison of machine learning technique

![AUC Score-ROC Curve]

Figure 28. AUC score-ROC curve

5. CONCLUSION

Analyzing various aspects of various models might lead to the following conclusions. When compared to the others, the KNN model has the highest accuracy of 88.52% with the value of k=5 for the selected dataset. The AUC value varies from 0.8-0.9, with the highest value being that of the KNN classifier. The second-best model is Random Forest with an accuracy of 86.89%. This research is beneficial to the medical profession since it allows both doctors and patients to continue therapy in the best possible way by providing an early and thorough understanding of a problem or condition. It is
simple to comprehend and has a lot of potential for improving sickness prediction. The work has the potential to be expanded further.

6. References

[1] Reddy KS, Shah B, Varghese C, Ramadoss A. 2005 Responding to the threat of chronic diseases in India The Lancet 366(9498) 1744-49 DOI: 10.1016/s0140-6736(05)67343-6

[2] Abdeldjouad FZ, Brahami M, Matta N. 2020 The Impact of Digital Technologies on Public Health in Developed and Developing Countries (LNCS vol 12157) ed maiel M and Mokhtari M et al (Cham: Springer) pp 299-306 DOI: 10.1007/978-3-030-51517-1_26

[3] Pahwa K, Kumar R. 2017 Prediction of heart disease using hybrid technique for selecting features 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical Computer and Electronics (UPCON) 500-04 DOI: 10.1109/upcon.2017.8251100

[4] Pouriyeh S, Vahid S, Sannino G, De Pietro G, Arabnia H, Gutierrez J. 2017 A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease 2017 IEEE Symposium on Computers and Communications (ISCC) 204-07 DOI: 10.1109/isc.cc.2017.8024530

[5] Mahboob T, Irfan R, Ghaffar B. 2017 Evaluating ensemble prediction of coronary heart disease using receiver operating characteristics 2017 Internet Technologies and Applications (ITA) IEEE 110-15 DOI: 10.1109/itecha.2017.8101920

[6] Bashir S, Qamar U, Javed MY. 2014 An ensemble based decision support framework for intelligent heart disease diagnosis International conference on information society (i-Society 2014) 259-64 DOI: 10.1109/i-society.2014.7009056

[7] Ashraf M, Ahmad SM, Ganai NA, Shah RA, Zaman M, Khan SA, Shah AA. 2021 International Conference on Innovative Computing and Communications (AISC vol 1165) ed Gupta D and Khanna A et al (Singapore: Springer) pp 229-255 DOI: 10.1007/978-981-15-5113-0_18

[8] David H, Beley SA. 2018 Heart disease prediction using data mining techniques ICTACT Journal on Soft Computing 9(1) 1817-23 DOI: 10.21917/ijsc.2018.0253

[9] Kahramanli H, Allahverdi N. 2008 Design of a hybrid system for the diabetes and heart diseases Expert systems with applications 35(1-2) 82-89 DOI: 10.1016/j.eswa.2007.06.004

[10] Palaniappan S, Awang R. 2008 Intelligent heart disease prediction system using data mining techniques 2008 IEEE/ACS international conference on computer systems and applications 108-15 DOI: 10.1109/aicesa.2008.4493524

[11] Andreotti F. et al 2020 Prediction of the onset of cardiovascular disease from electronic health records using multi-task gated recurrent units arXiv (preprint arXiv:2007.08491)

[12] UCI-Heart Disease Data Set [Online] Available: https://www.kaggle.com/ronitf/heart-disease-uci (Accessed on May 1, 2021)

[13] Mathuria M, Bhargava R, Sharma G, Bhargava N. 2013 Decision tree analysis on j48 algorithm for data mining International Journal of Advanced Research in Computer Science and Software Engineering 3(6) 1114-19

[14] Gao XY, Amin Ali A, Shaban Hassan H, Anwar EM. 2021 Improving the Accuracy for Analyzing Heart Diseases Prediction Based on the Ensemble Method 2021 DOI: 10.1155/2021/6663455

[15] Ananey-Obiri D, Sarku E. 2020 Predicting the Presence of Heart Disease using Comparative Data Mining and Machine Learning Algorithms International Journal of Computer Applications 176 17-21 DOI: 10.5120/ijca2020920034