Heart Disease Prediction using Machine Learning algorithms

Ismael Reséndiz Robles

Texas State University - CS 7317

Contents

1 Introduction

Based on the Center for Disease Control and Prevention, heart disease is the leading cause of death for men, women, and people of most racial and ethnic groups in the United States. One person dies every 36 seconds in the United States from cardiovascular disease. About 655,000 Americans die from heart disease each year, that's 1 in every 4 deaths. Heart disease costs the United States about $219 billion each year from 2014 to 2015. This includes the cost of health care services, medicines, and lost productivity due to death [4].

This makes heart disease a major concern to be dealt with. But it is difficult to identify heart disease because of several risks factors such as age, high blood pressure, high cholesterol and many other. This is the reason researchers have turn their attention to new methods to model these kinds of problems like Machine Learning algorithms.

Machine learning(ML) model predictions allow businesses to make highly accurate guesses as to the likely outcomes of a question based on historical data. There are multiple heart datasets available on Internet. For this project I have selected the Clevelant Heart Disease dataset[6] since it is the most refined and there is a wide range of results posted online which can be used to compare against the output of this project.

The purpose of this project is to explore existing papers related to solve heart disease prediction problem and to experiment with the top most efficient ML algorithms in search of improvement opportunities. Compare those algorithms and provide an insight of the accuracy, sensitivity and efficiency towards this problem.

2 Related Work

There are articles that study the prediction of heart diseases using Artificial Neuronal Networks (ANN), logistic regression, decision tree, ensemble model and others. In [7] the authors proposes a prediction model based on ensemble, which combines three independent models Support Vector Machine (SVM), decision tree and ANN to improve the accuracy of each independent model to 87%. The authors in [2] discuss an approach to select the best features from the dataset using Principal Component Analysis (PCA), and apply them to a logistic regression algorithm on which tuning techniques have been implemented and resulted in 100% accuracy.

3 Development Plan

The development plan of this project will consist of 3 phases.

Research
This phase is to acquire current state of the problem, learn about the progress made in this field and the methodologies utilized to solve this or similar tasks. As well as to review datasets and practises to gain knowledge, and use it towards this project.
Estimated completion date is Oct 25, 2020.

Implementation
After the research is done, the information collected would serve as a guide of what ML algorithms and techniques to implement and experiment with. In this phase, the motivation is to try a variation of techniques, parameters or combinations to pursuit improvement in the existing algorithms accuracy and sensitivity.
Estimated completion date is Nov, 10 2020.

Analyze and Report
In the last phase, the results of the experiment will be analyzed: a detailed comparison of the algorithms implemented to show their effectiveness and derive their properties.
Estimated completion date is Nov 18, 2020.

4 Problem Description

ML algorithms can assist us in making decisions and predictions based on mathematical models. ML offers a variety of techniques and algorithms that have different properties and can work more adequately on one type of problem, but the same technique might not work on a different problem. In this project, I will be applying different ML approaches to the heart disease dataset, and compare them using evaluation metrics and classification accuracy.

4.1 Data Set

The dataset used in this article is the Cleveland Heart Disease dataset taken from the UCI repository. This database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. This subset contains the most important predictors and factors to determine the disease. For example, and to mention few of these features. *Age*: Studies indicate that the risk of stroke doubles every decade after *age* 55. Other studies show that men have higher risk of coronary disease, however females with diabetes is more likely to develop a heart disease than men. High blood pressure overtime damages the arteries in the human body, in combine with diabetes or high cholesterol the chances of developing a heat disease are even higher. The objective of the project is not to refine the features on the dataset, but to focus on the problem using work from previous papers. Then analyze the result given by different ML algorithms and to strive for areas of improvement and result which can then be leveraged in future works.

Below is detailed description of each feature:

1. Age of the patient in years (AGE).
2. Sex of the patient (SEX).
   1 = male; 0 = female
3. Chest Pain Type (CP).
   1 = typical angina; 2= atypical angina; 3= non-anginal pain; 4= asymptomatic
4. Resting Blood Pressure (TRESTBPS) in mmHg.
5. Serum Cholesterol in mg/dl (CHOL).
6. Fasting Blood Sugar greater than 120 mg/dl (FBS).
   1 = true; 0 = false
7. Resting Electrocardiograph results(REST_ECG).
   0 = normal; 1 = having STT wave abnormality; 2 = showing probable or definite left ventricular hypertrophy by Estes criteria
8. Maximum Heart Rate Achieved in bps(THAL_ACH).
9. Exercise Induced Angina (EXANG). *1 = yes; 0 = no*
10. ST depression induced by exercise relative to rest (OLD_PEAK).
11. The slope of the peak exercise ST segment (SLOPE).
    1 = upsloping; 2 = flat; 3 = downsloping
12. Number of major vessels (0-3) colored by flourosopy (CA).
13. Thalassemia (THAL).
    3 = normal; 6 = fixed defect; 7 = reversible defect
14. The predicted attribute (PRED).
    1 = true; 0 = false

4.2 Prediction Models

As stated in previous sections, this project will consist in applying supervised ML models, evaluate them and intend to derive their properties in search of improving their accuracy. I will be implementing these ML models in python using the **sklearn**[3] and **keras**[5] libraries. The following classification models will be implemented:

- Logistic Regression is a type of regression analysis in statistics used for prediction which uses a logistic function to produce an output between 0 and 1.

$$logistic(z) = \frac{1}{1 + exp(-z)}$$

- Decision Tree is another ML algorithm that offers great classification accuracy with less computational power. Decision Tree's goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.
- Random Forest are made of many decision trees, on which there exist two key concepts that gives it the name random:

  - Random sampling of training data points when building trees
  - Random subsets of features considered when splitting nodes

  The idea is that by training each tree on different samples, although each tree might have high variance with respect to a particular set of the training data, overall, the entire forest will have lower variance but not at the cost of increasing the bias.

- Artificial Neural Network (ANN) can be use for patter recognition or data classification through a learning process. ANN can perform task that a linear classifier can not, the main advantage is the capacity to find complex relations among features, with high tolerance to data uncertainty, and predicting patterns with high accuracy.

5 Results

In this section, the result of each model will be evaluated by their accuracy, using the confusion matrix and statistically.
By using the confusion matrix we can extract exactly the following counts:

- True positive (TP): It's the model outcome where correctly predicts the positive class.
- True negative (TN): It's the model outcome where correctly predicts the negative class.
- False Positive (FP): It's the model outcome where incorrectly predicts the positive class.
- False Negative (FN): It's the model outcome where incorrectly predicts the negative class.

Given these metrics we can then calculate the model sensitivity and specificity as below:

$$Sensitivity = \frac{TP}{TP + FN}$$

| True positive rate.

$$Specificity = \frac{TN}{TN + FP}$$

| False positive rate.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: True Positive Rate. False Positive Rate and it's useful to understand the trade-off in the true-positive and false-positive rate for different thresholds.
Another metric commonly use is the Geometric mean (G-Mean), which seeks to balance the model between sensitivity and specificity. I will use G-Mean to evaluate the model classification balance.

$$G - Mean = \sqrt{Sensitivity * (1 - Specificity)}$$

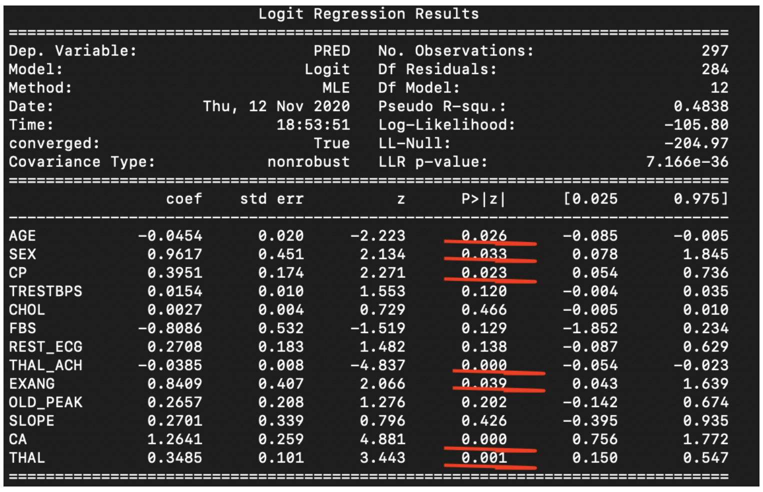
The dataset is divided in the training set 80% and testing set 20%.

5.1 Logistic Regression

From the initial logistic regression formula, we can input all the features from the experiment to illustrate the model.

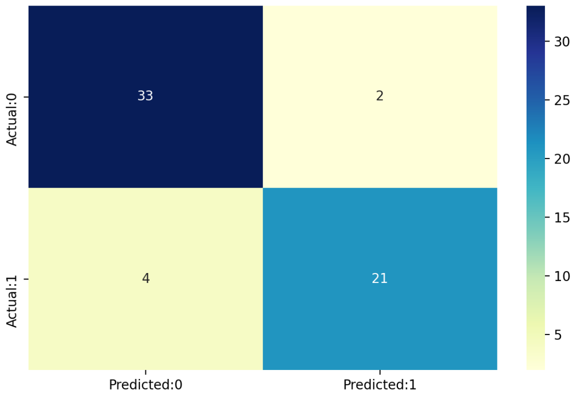
$$P(y = 1) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + ... + \beta_n x_n))}$$

Due to the number of features and the dataset size of this problem, the model is setup with the Limited-memory BFGS [1] algorithm as optimizer.
The feature selection output from the logistic regression are in the below table:



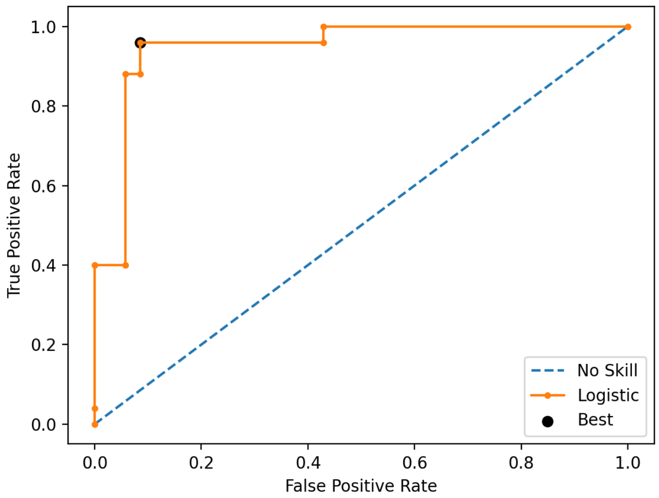
```
                           Logit Regression Results
==============================================================================
Dep. Variable:                   PRED   No. Observations:                  297
Model:                          Logit   Df Residuals:                      284
Method:                           MLE   Df Model:                           12
Date:                Thu, 12 Nov 2020   Pseudo R-squ.:                  0.4838
Time:                        18:53:51   Log-Likelihood:                -105.80
converged:                       True   LL-Null:                       -204.97
Covariance Type:            nonrobust   LLR p-value:                 7.166e-36
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
AGE           -0.0454      0.020     -2.223      0.026      -0.085      -0.005
SEX            0.9617      0.451      2.134      0.033       0.078       1.845
CP             0.3951      0.174      2.271      0.023       0.054       0.736
TRESTBPS       0.0154      0.010      1.553      0.120      -0.004       0.035
CHOL           0.0027      0.004      0.729      0.466      -0.005       0.010
FBS           -0.8086      0.532     -1.519      0.129      -1.852       0.234
REST_ECG       0.2708      0.183      1.482      0.138      -0.087       0.629
THAL_ACH      -0.0385      0.008     -4.837      0.000      -0.054      -0.023
EXANG          0.8409      0.407      2.066      0.039       0.043       1.639
OLD_PEAK       0.2657      0.208      1.276      0.202      -0.142       0.674
SLOPE          0.2701      0.339      0.796      0.426      -0.395       0.935
CA             1.2641      0.259      4.881      0.000       0.756       1.772
THAL           0.3485      0.101      3.443      0.001       0.150       0.547
==============================================================================
```

After creating the model and predicting the classification with the test data, the accuracy obtained in this model is 0.9 (90%). Below is the confusion matrix and statistics.

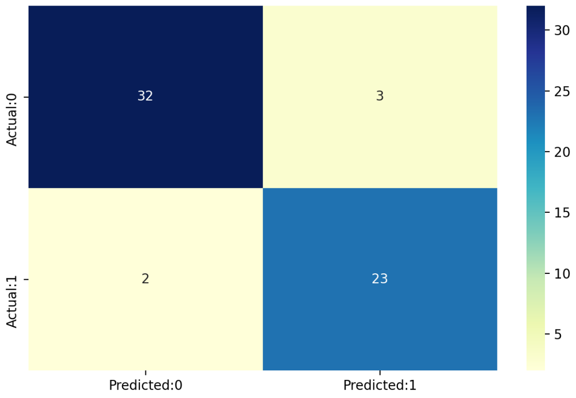


Accuracy	$21 + 33/21 + 33 + 2 + 4 = 0.9$
Sensitivity	$21/21 + 4 = 0.84$
Specificity	$33/33 + 2 = 0.94$

Above results show the model operating with a threshold of 0.5 i.e. the model decides weather the patient has a heart disease if the outcome probability is greater than 0.5, else the patient is healthy. However, there might be an opportunity to calculate a better threshold for the model using the G-Mean metric. First I will plot the ROC curve to obtain the specificity and sensitivity rates. Then, I will use these rates to calculate each G-Mean and finally select the largest value. The results show the largest G-Mean is 0.937 and its corresponding threshold is found at 0.36. Testing the model against this threshold, the



result show a small reduction on the false-negative count, however a unit increase in the false-positive count. Plotting the confusion matrix again with the new test results we obtain an accuracy of 0.916 (91%). A slight improvement from the initial model.



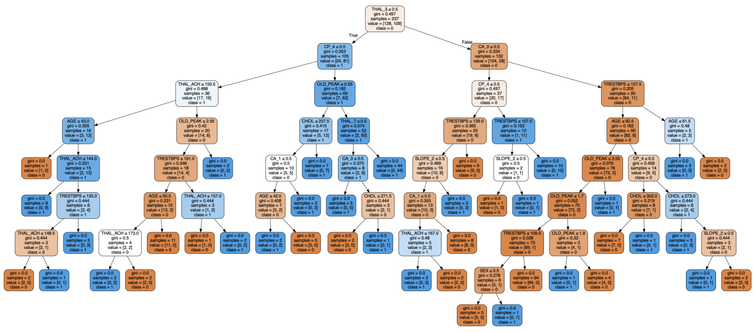
5.2 Decision Tree

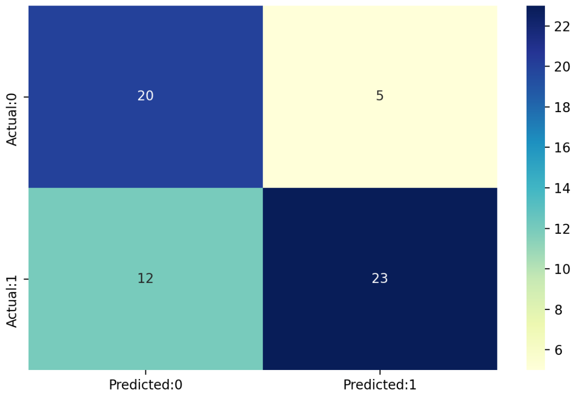
I opted for decision trees due to their flexibility and compatibility in classification problems. Beginning to build a top-down tree, and deciding on which features to choose and what conditions to use for splitting, along with knowing when to stop. To avoid an arbitrarily tree growth, I will test few approaches to limit the tree depth size, number of samples to split a node and pruning.

Initializing the tree and set the training data and no boundaries, result in a 8 level tree and 0.71(71%) classification accuracy. Additionally this model selected the below features as most important over all the nodes that are split on that feature.

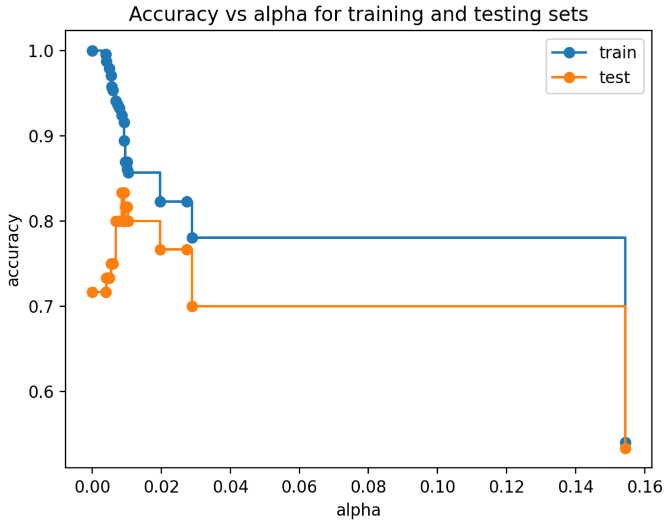
Feature	**Importance**
Normal Thalassemia level (THAL_0)	0.412
Asymptomatic Chest Pain(CP_4)	0.156
Age (AGE)	0.086
Maximum Heart Rate Acchived (THAL_ACH)	0.079
No major vessels (CA_0)	0.070

Below is the complete tree and the model confusion matrix.

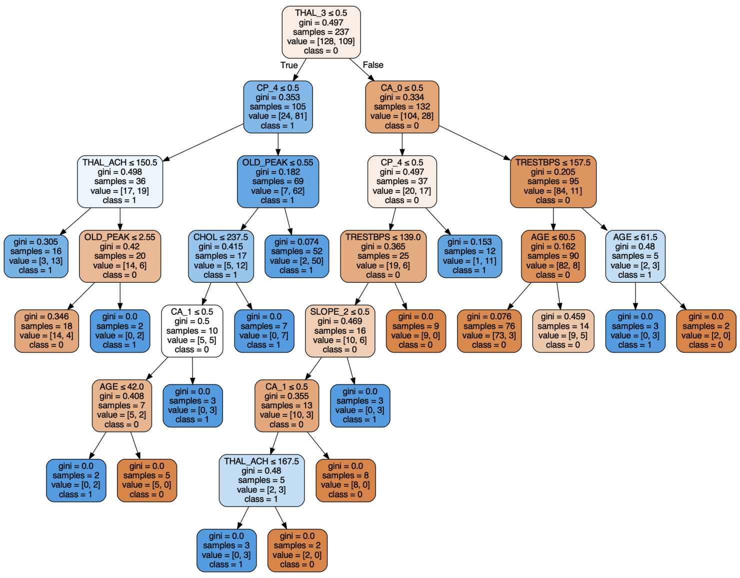


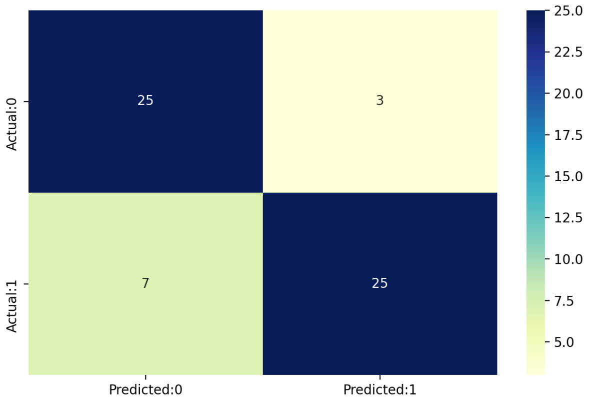


Next, I will set random parameters to reduce the size if the tree and avoid overfiting. For this, I will use Cost complexity pruning, by plotting the leaves impurities vs the effective alphas, and then capture the most effective alpha value which produces a most accurate model.



from the graph above, the best alpha value is 0.0084. This parameter when set in the model the accuracy improves to 0.83 (83%). Below is the pruned decision tree and the updated confusion matrix.

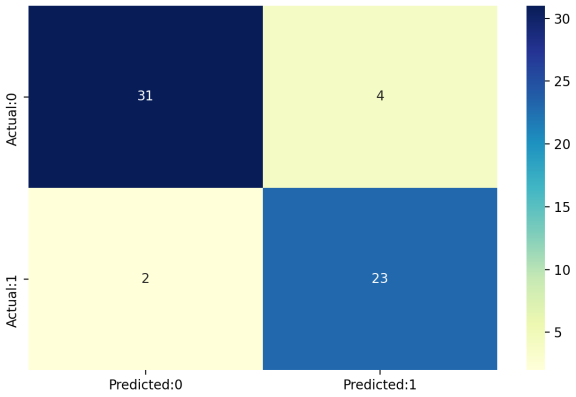


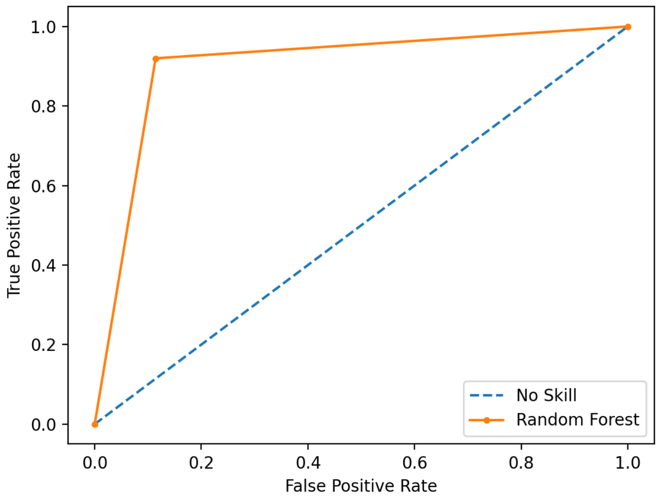


5.3 Random Forest

Since the decision tree performs well on training data but not on the testing data, this indicates that there are some level of overfiting, which can happen when the model memorizes the training data by fitting it closely. The problem is that the model learns not only the actual relationships in the training data, but also any noise that is present. Give that random forest combines several decision trees, trains each one differently and the final predictions are made by averaging the predictions of each individual tree, the expected accuracy should be higher than a single decision tree.

initializing the random forest with a set of 100 estimators (decision tree classifiers) and fitting the training data, the accuracy obtained is 0.9(90%). Below is the confusion matrix and ROC curve.

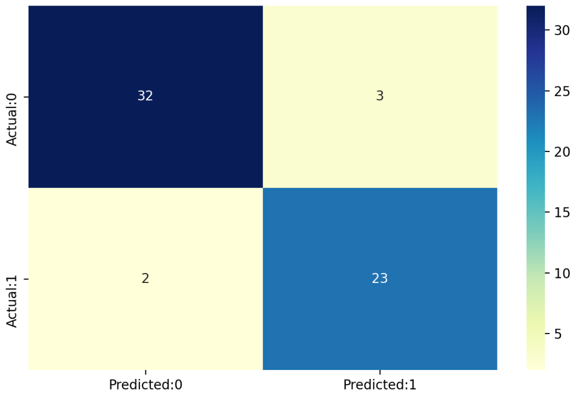




And the list of best features detected by the model are below

Feature	**Importance**
Maximum Heart Rate achieved ($THAL_0$)	0.118
No major vessels (CA_0)	0.102
ST depression (OLD_PEAK)	0.093
Age (AGE)	0.088
Cholestereol level (CHDL)	0.076

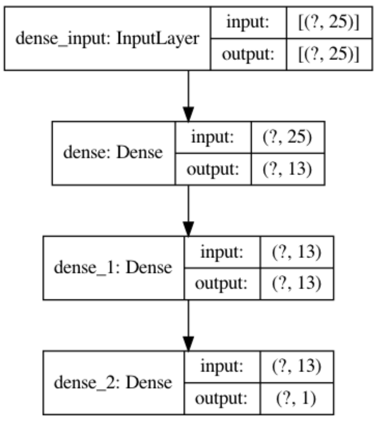
By adjusting the number of estimator to 150 and 200, the model accuracy slightly increases to 0.916 (91.6%).



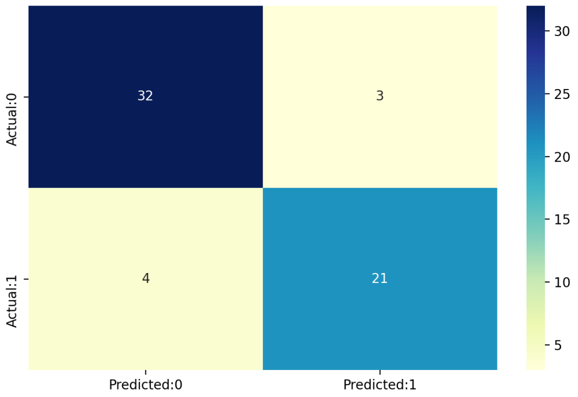
5.4 Artificial Neural Network (ANN)

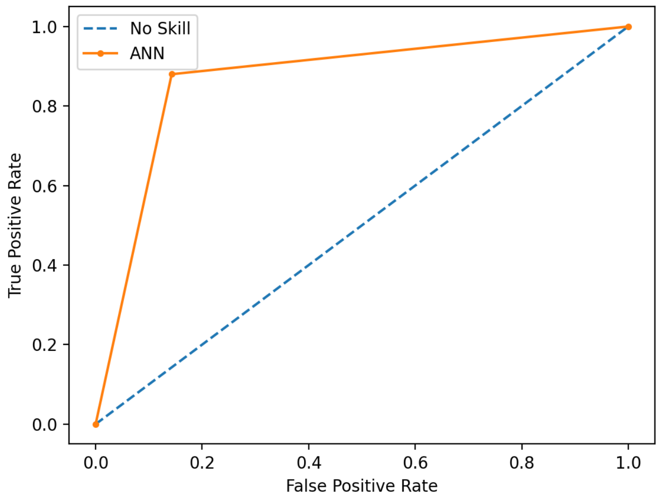
A feed forward network should be enough to receive all input features for this problem. Since the dataset contains some categorical variables, those columns need to be transformed. For example Chest Pain contains 4 values, then an additional 3 new columns are needed to map all possible values. After this transformation the number of columns are 25. Subsequently, a level of normalization can be applied to the data in order the help the neural network to process it faster.

Afterwards, the model is constructed with a dense layer of 25 neurons to receive the input data, which activation formula is ReLU. Following by another dense layer to reduce the number of neurons to the mean value of total neurons (25 +1) = 13. Finally, a single neuron layer with Sigmoid activation to output the classification result from 0 to 1. the below picture depicts the model in a high level.



As per literature, best option for binary classification are usually binary cross entropy, and the optimizer algorithm is Adam. Additionally, the model is set with a 8 batch size and 100 epochs.
The accuracy obtained in the model is 0.916 (91.6%).





6 Conclusions

6.1 Outcome comparison

The results obtained in this project show the performance in terms on classification accuracy on various machine learning algorithms. The Logistic regression model, Random Forest can perform as good as ANN with some parameter updates.
Dataset is small and requires level of treatment or normalization before feeding it to the models. Below is a compare of the results from each model.

Model	**Accuracy**	**Sensitivity**	**Specificity**
Logistic Regression	0.9	0.84	0.94
Logistic Regression (Hyper Parameter tweak)	0.916	0.884	0.941
Decision Tree	0.71	0.821	0.625
Decision Tree (Alpha tweak)	0.83	0.892	0.781
Random Forest	0.90	0.851	0.939
Random Forest (Forest Size tweak)	0.916	0.92	0.941
ANN	0.916	0.875	0.888

6.2 Future Work

- Search for optimization opportunities in the prediction models.
- Test the models with different datasets.
- Build a system that uses the prediction models to predict with user input data.

References

[1] S. Ambesange, V. A, S. S, Venkateswaran, and Y. B S. *Limited-Memory Broyden-Fletcher-Goldfarb-Shanno Algorithm in ML.NET.*

[2] S. Ambesange, V. A, S. S, Venkateswaran, and Y. B S. Multiple heart diseases prediction using logistic regression with ensemble and hyper parameter tuning techniques. In *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, pages 827–832, 2020.

[3] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[4] CDC. *Heart Disease Facts*, 2020. https://www.cdc.gov/heartdisease/facts.htm.

[5] Francois Chollet. *Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow.*

[6] UCI. *University of Clevelant*, 1988. https://archive.ics.uci.edu/ml/datasets/Heart+Disease.

[7] X. Wenxin. Heart disease prediction model based on model ensemble. In *2020 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pages 195–199, 2020.