Technical Report

Applying classification techniques for heart disease detection.

By: Panagiotis Stenos

Contents

List of Figures

1. Aim of the Project

The aim of this project is to develop a classification algorithm that identifies individuals with heart disease based on some of their health condition and other relevant attributes. To achieve that several classifiers were trained on a labelled dataset of patients, as well as ensemble methods, boosting and stacking methods. Results from all these methods were thoroughly analyzed and the advantages and disadvantages of each of the methods used were rigorously discussed. The classifier that was ultimately selected was the one that achieved the best balance between performance and interpretability.

2. Introduction

Cardiovascular diseases (CVDs) are the number 1 cause of death globally, taking an estimated 17.9 million lives each year, which accounts for 31% of all deaths worldwide. Four out of 5CVD deaths are due to heart attacks and strokes, and one-third of these deaths occur prematurely in people under 70 years of age. Heart failure is a common event caused by CVDs and this dataset contains 11 features that can be used to predict a possible heart disease.

People with cardiovascular disease or who are at high cardiovascular risk (due to the presence of one or more risk factors such as hypertension, diabetes, hyperlipidemia or already established disease) need early detection and management wherein a machine learning model can be of great help.

3. Data Collection

This dataset was obtained from Kaggle and created by combining different datasets already available independently but not combined before. In this dataset, 5 heart datasets are combined over 11 common features which makes it the largest heart disease dataset available so far for research purposes. The five datasets used for its curation are:

- Cleveland: 303 observations
- Hungarian: 294 observations
- Switzerland: 123 observations
- Long Beach VA: 200 observations
- Stalog (Heart) Data Set: 270 observations
- Total: 1190 observations
- Duplicated: 272 observations

Final dataset: 918 observations (12 features)

Attribute information:

1. Age: age of the patient [years]
2. Sex: sex of the patient [M: Male, F: Female]
3. ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
4. RestingBP: resting blood pressure [mm Hg]
5. Cholesterol: serum cholesterol [mm/dl]
6. FastingBS: fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
7. RestingECG: resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]

8. MaxHR: maximum heart rate achieved [Numeric value between 60 and 202]
9. ExerciseAngina: exercise-induced angina [Y: Yes, N: No]
10. Oldpeak: oldpeak = ST [Numeric value measured in depression]
11. ST_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]
12. HeartDisease: output class [1: heart disease, 0: Normal]

4. Exploratory Data Analysis

Exploratory data analysis is a necessary step that helps the data scientist in gaining insights into the data distribution, identifying patterns, spotting outliers and understanding the characteristics of each feature, which can inform preprocessing decisions and model selection during the subsequent modelling phase. The figure below shows the distribution of all the attributes of the dataset.

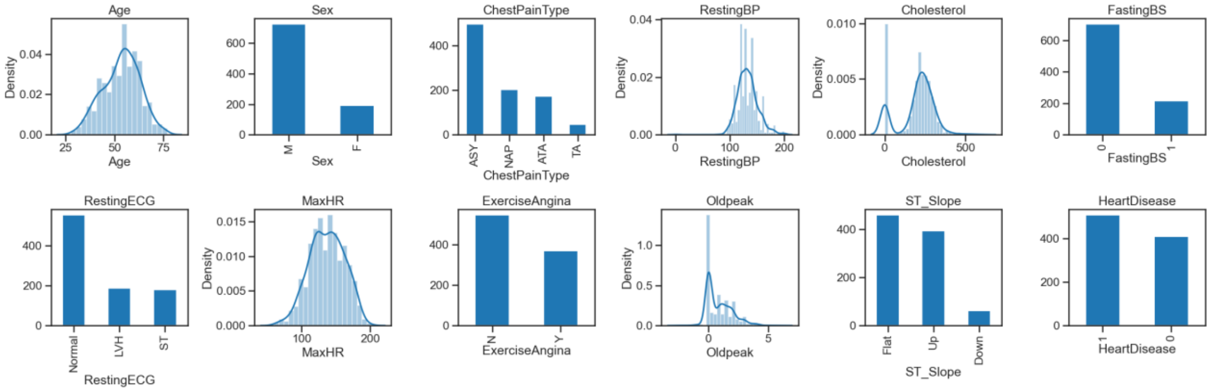

Figure 1: Distribution of attributes

After visualizing the feature distributions, I observed an anomaly in the 'Cholesterol' distribution. Notably, around 20% of the dataset, or 172 data points, had a cholesterol value of zero, which is implausible given that individuals always have some level of cholesterol. Furthermore, the second lowest cholesterol value was 85, confirming my hypothesis that the zero values are likely errors or missing data in the dataset. There are various methods to address this issue. While substituting with the mean or another statistic is a legitimate approach, considering the high-risk health domain of the data, I opted to eliminate all rows with zero cholesterol. The graph below illustrates the distributions of 'Cholesterol' and ‘HeartDisease’ features after the elimination of the zero cholesterol rows.

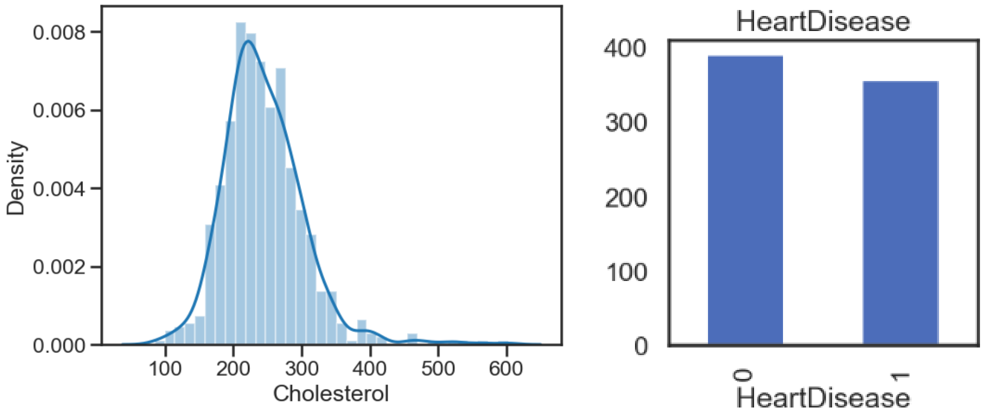

Figure 2: Modified distribution of 'Cholesterol' and ‘HearDisease’

It is noteworthy to observe the distribution of the target class post-modification. The predominant class now pertains to patients without heart disease, contributing to a generally balanced distribution across classes. This balance achieved is ideal for modelling as classifiers tend to optimize their performance by reducing accuracy error; and hence often perform poorly on underrepresented classes.

5. Modelling

In this section, supervised machine learning algorithms were optimized on the training dataset and their performance was evaluated. The metrics used for the evaluation of the models were: a. accuracy score, b. recall, c. area under the ROC (receiver and operating characteristics) curve, d. precision, and e. F1-score. All those metrics were calculated using the testing dataset. The figure below visualizes the performance of the top-performing classifier from each of the Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Decision Tree models. See the python notebook for more information.

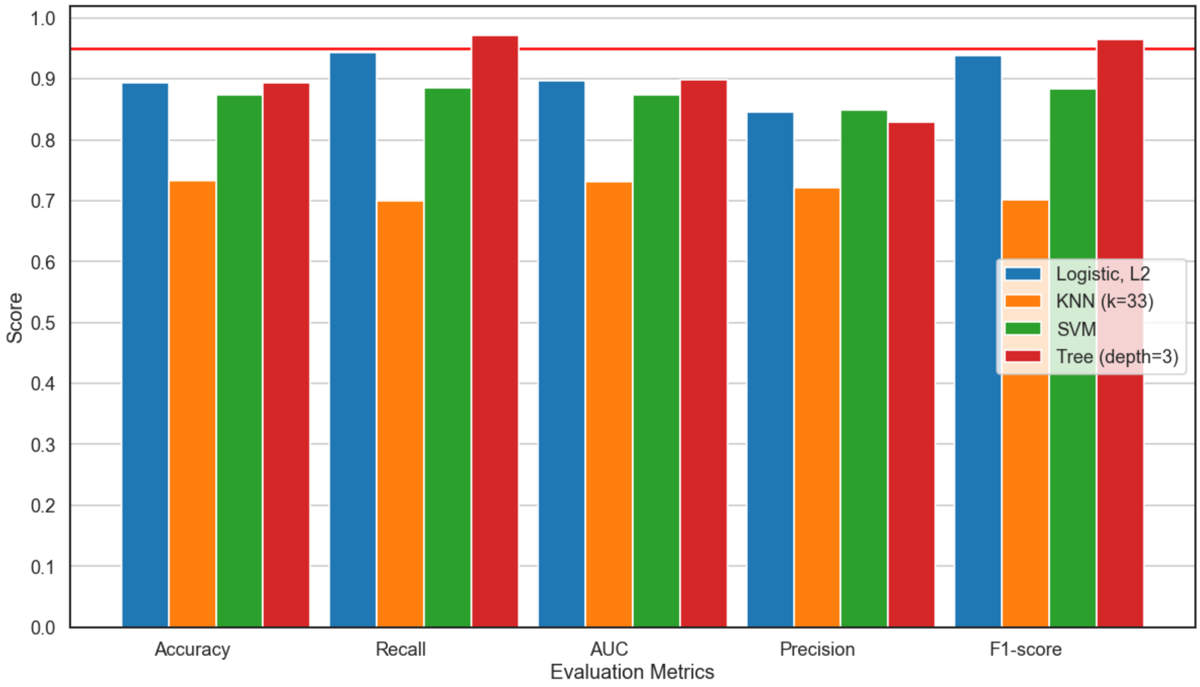

Figure 3: Performance of different classifiers

The best performing model out of the four is undoubtedly decision trees (with a depth of 3 and using 'gini' as the criterion for splitting the leafs). Other decision tree models were tried out (with higher depth) but were found to be overfitting on the training data; hence, showing poor performance on the testing data.

For the logistic regression classifier, two models were evaluated: one incorporating L1 regularization error, and the other incorporating L2 regularization error. All KNN models tested exhibited subpar performance, while the SVM demonstrated a fair overall performance.

Various ensemble methods based on those classifiers were explored in an effort to improve testing accuracy. Three ensemble methods were used: Bagging (bootstrap aggregating), Boosting and

Stacking. For more information on how these methods work check the python notebook. The figure below visualizes the performance of the top-performing ensemble models.

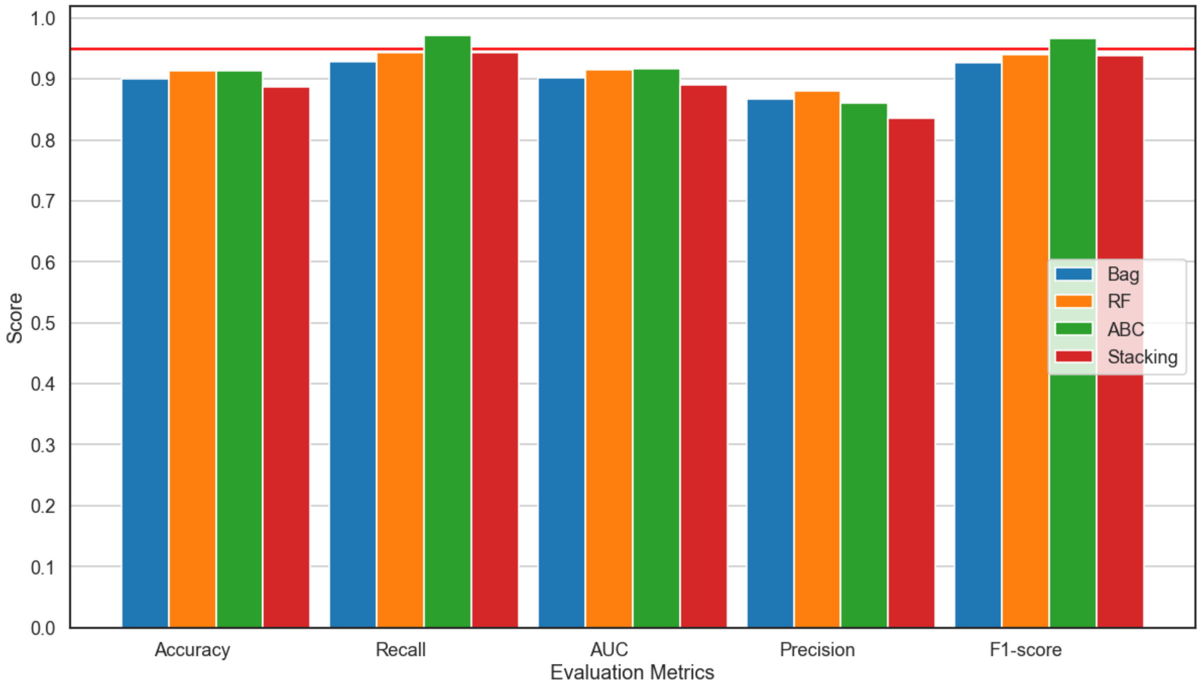

Figure 4: Performance ensemble models

All four ensemble models (Bagging, Random Forest, Adaptive Boosting Classifier and Stacking Classifier) show high performance. AdaBoost performs slightly better than all the other classifiers, with only a slightly lower precision than Random Forest.

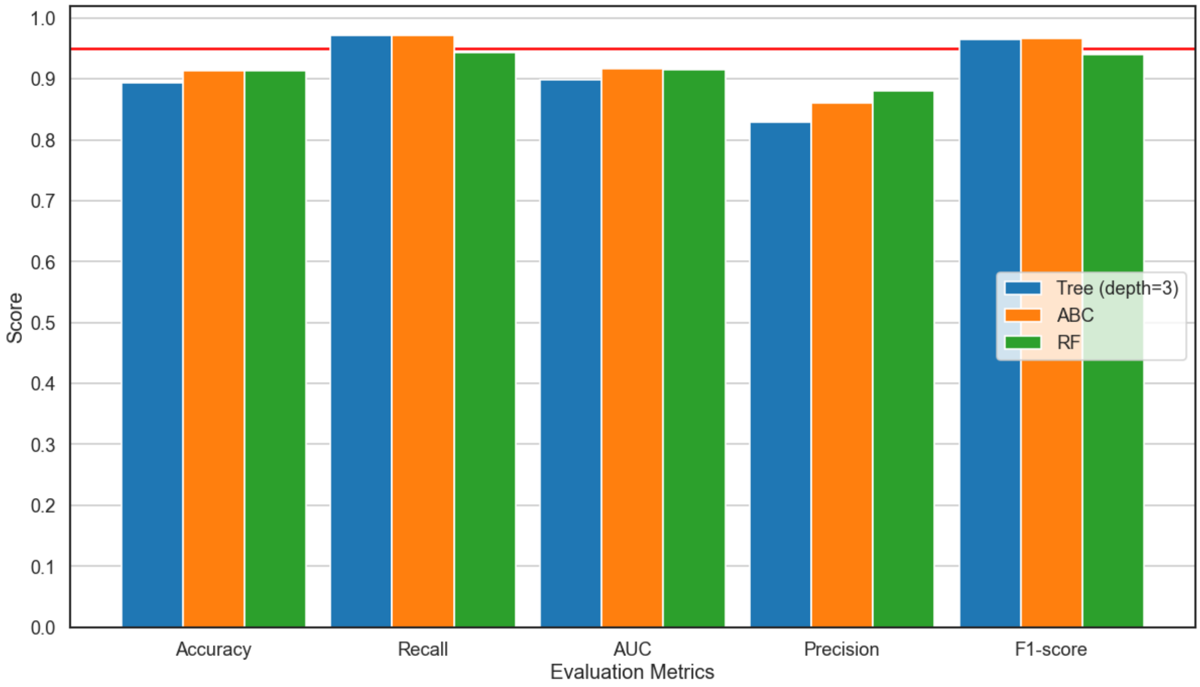

Figure 5: Best performing classifiers

6. Model Evaluation

After modelling several algorithms, AdaBoost stands out as the top performer across various metrics, showcasing superior accuracy, recall, and F1 score. Random Forest exhibits a slightly superior precision metric when compared to AdaBoost, although it lags behind in the remaining performance metrics. Interestingly, the Decision Tree algorithm closely trails AdaBoost, delivering comparable results.

Model performance is undoubtedly crucial, but in high-risk sectors such as healthcare, working with an easily interpretable model is crucial as consequences of decisions hold significant weight. Decision tree is a self-interpretable model and can provide a transparent view of the decision-making process. Unlike ensemble models that may resemble "black boxes," Decision tree allows stakeholders to follow each branch of the tree, comprehending how input features contribute to the final prediction. This transparency is particularly valuable in this case where a medical professional will be able to follow the logic of the tree and confirm the validity of the model from a scientific point of view (or disprove it due to a biased dataset), fostering confidence among other healthcare professionals and end-users. The straightforward representation of decision logic in decision trees enhances communication and collaboration between data scientists and domain experts, a crucial aspect in refining models for real-world applications in the health sector. Therefore, while AdaBoost demonstrates remarkable performance, the Decision Tree algorithm emerges as a compelling alternative due to its nearly equivalent effectiveness and self-interpretable nature. Achieving a balance between performance and interpretability is paramount, particularly when dealing with sensitive health data.

The classification model selected, decision tree, while powerful and interpretable, has certain potential flaws that should be considered. One notable limitation is decision trees' susceptibility to overfitting, especially when the tree is deep. Because of the way trees are modelled, small changes in the data (or noise) will lead to different decision boundaries. Tree will therefore generalize poorly on unseen data. If errors rise on unseen data, some of the ensemble classifiers modelled might be preferred, especially bagging models as they generalize very well on unseen data.

7. Results

Table 1: Decision boundaries of the decision tree algorithm

```
|--- feature_10 <= 1.50
|   |--- feature_1 <= 0.50
|   |   |--- feature_8 <= 0.50
|   |   |   |--- class: 0
|   |   |--- feature_8 >  0.50
|   |   |   |--- class: 1
|   |--- feature_1 >  0.50
|   |   |--- feature_7 <= 150.50
|   |   |   |--- class: 1
|   |   |--- feature_7 >  150.50
|   |   |   |--- class: 1
|--- feature_10 >  1.50
|   |--- feature_2 <= 0.50
|   |   |--- feature_8 <= 0.50
|   |   |   |--- class: 0
|   |   |--- feature_8 >  0.50
|   |   |   |--- class: 1
|   |--- feature_2 >  0.50
|   |   |--- feature_9 <= 2.25
|   |   |   |--- class: 0
|   |   |--- feature_9 >  2.25
|   |   |   |--- class: 1
```

```
|--- ST_Slope == [Flat, Down]
|   |--- Sex == Female
|   |   |--- ExerciseAngina == No
|   |   |   |--- class: 0
|   |   |--- ExerciseAngina == Yes
|   |   |   |--- class: 1
|   |--- Sex == Male
|   |   |--- MaxHR <= 150.50
|   |   |   |--- class: 1
|   |   |--- MaxHR >  150.50
|   |   |   |--- class: 1
|--- ST_Slope == Up
|   |--- ChestPainType == ASY
|   |   |--- ExerciseAngina == No
|   |   |   |--- class: 0
|   |   |--- ExerciseAngina == Yes
|   |   |   |--- class: 1
|   |--- ChestPainType == [ATA, NAP, TA]
|   |   |--- Oldpeak <= 2.25
|   |   |   |--- class: 0
|   |   |--- Oldpeak >  2.25
|   |   |   |--- class: 1
```

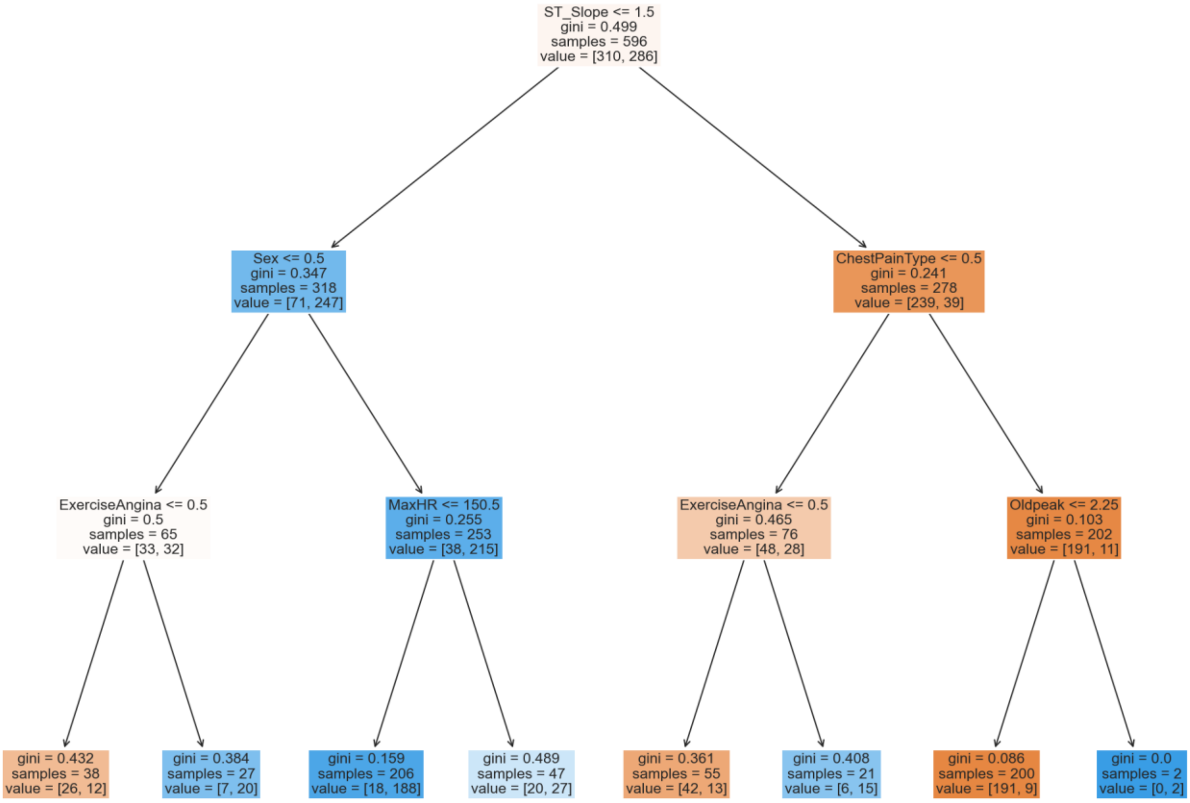

Figure 6: Tree diagram