Heart Disease Prediction Using Machine Learning

Ran An
Data Science Initiative at Brown University
December 7, 2022

Introduction

I will explore the prediction of heart disease in this project because heart disease is the leading cause of death globally (WHO, 2021). Efficient early detection methods play a role in helping medical experts recognize patients at high risk of having heart disease. If we can predict whether a patient has heart disease, then patients can receive treatment early and live a healthier life. I will use various machine learning models to predict heart disease. This is a classification problem that uses 11 input features to predict the binary target variable that represents the presence of heart disease in patients.

I'm using the Heart Failure Prediction dataset from Kaggle, which combines 5 heart datasets over 11 common features from UCI Machine Learning Repository. This dataset contains 918 observations and 12 columns (11 features and 1 target variable). The 11 features of the dataset represent the attribute information of each patient. There are 5 numerical features: Age, RestingBP (resting blood pressure), Cholesterol, MaxHR (maximum heart rate achieved), and Oldpeak. The 6 categorical features include Sex, ChestPainType, FastingBS (fasting blood sugar), RestingECG (resting electrocardiogram results), ExerciseAngina (whether the patient has exercise-induced angina), and ST_Slope.

This dataset has been used in research and public projects. Patel et al. (2016) used its subset (Cleveland database with 303 instances and 76 attributes) to test the performance of decision tree algorithms for heart disease prediction. By comparing different algorithms, the authors found that the J48 algorithm achieves the highest prediction accuracy of 56.76%. The dataset I'm using combines 5 heart databases from UCI, one of which is the Cleveland database. The combined Kaggle dataset is used in many public projects. Menna-Tallah Nasr used 11 features in this dataset to predict whether a patient has heart disease and found that the XGBoost algorithm gives the best accuracy: 91% on training data, and 80% on test data.

Exploratory Data Analysis

The target variable is binary, taking values of 1 and 0. It turns out that the dataset is balanced with a balance of 0.55. Exploring the relationships between features and the target variable, I found that sex and age are strongly correlated with the presence of heart disease. I used a stacked bar plot to check the fraction of patients with heart disease in females and males, as shown in Figure 1. The plot shows that males have more chance to have heart disease. For the continuous variable, age, I used a box plot to see the distribution of ages in patients with heart disease and patients without heart disease, as shown in Figure 2. I found that the box for patients with heart disease is comparatively short and higher, suggesting that patients with heart disease tend to be older.

I also explored the relationships between features. The most correlated features are age and maximum heart rate. They are negatively correlated with each other as shown in Figure 3. By doing further research, I found that the maximum heart rate can be estimated by subtracting age from 220 (CDC, 2022). This aligns with the negative correlation shown in the scatterplot.

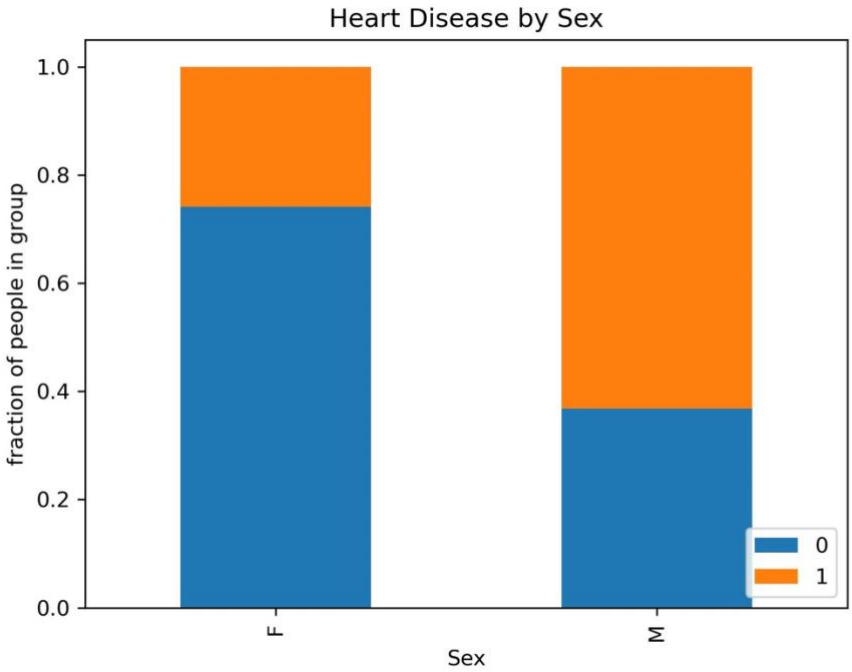

Figure 1. Males have more chance of having heart disease. The orange stacks represent heart disease while the blue stacks represent no heart disease. The male stack on the right has a larger portion of heart disease presence.

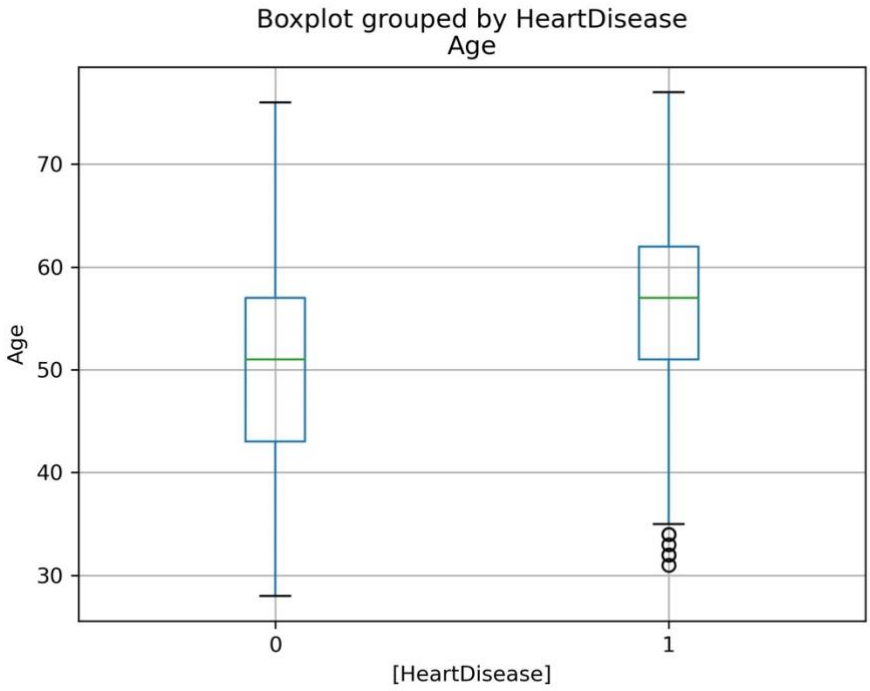

Figure 2. Age distribution for patients without vs with heart disease. Patients with heart disease (the right box) tend to be older.

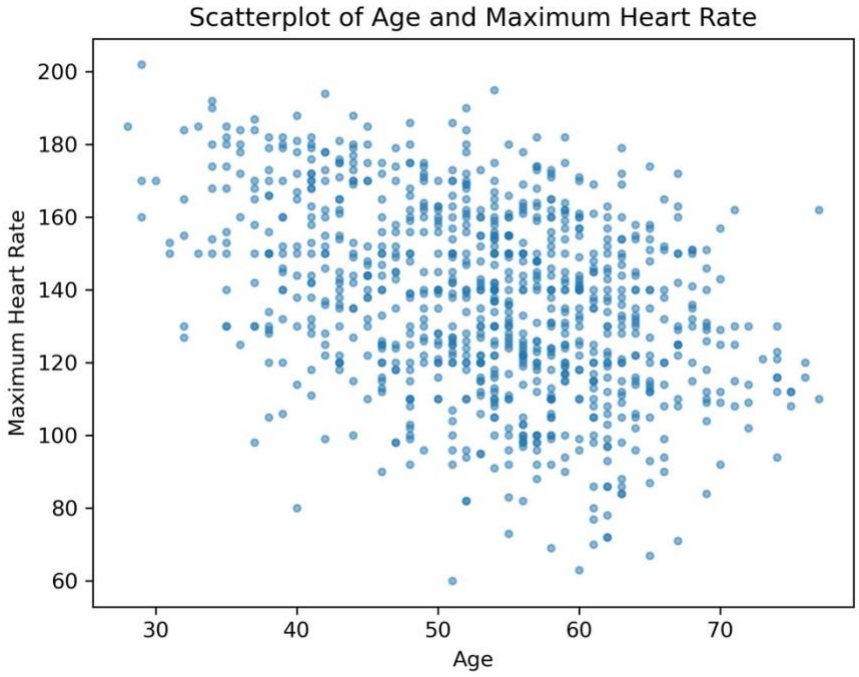

Figure 3. Relationship between age and the maximum heart rate. Each data point represents a patient. The data cluster shows a negative correlation between age and maximum heart rate.

Methods

The dataset is independent and identically distributed. It contains neither group structure, nor time series. Each observation represents an independent patient. Considering that the dataset is i.i.d. and balanced, I choose the basic split method and k-fold cross validation as my splitting strategy.

The dataset does not have missing values in any column. I applied MinMaxScaler to age because age is bounded between 28 and 77 in this dataset. Also, it approximately follows a normal distribution. For other continuous variables, I used StandardScaler as a preprocessor. After research, I found that the values of chest pain type and resting electrocardiogram results can be ordered in terms of severity. Thus, I applied OrdinalEncoder to these two features. OneHotEncoder is used for other categorical features without ranked values.

My machine learning pipeline's input includes the feature matrix, the target variable, the preprocessor, an initialized machine learning mode, and the model's hyperparameters I'd like to tune. I first use an 80-20 basic split to get the test set and the other set. Then I apply k-fold with 4 folds to the other set. After splitting the data, I use GridSearchCV with accuracy as the evaluation metric to tune the hyperparameters. The best model and best accuracy score are saved for further analysis.

Four machine learning algorithms are trained in this project to predict the presence of heart disease: logistic regression, random forest, support vector machine, and XGBoost. For each machine learning algorithm, I loop through 10 different random states. For each random state, my machine learning pipeline discussed above finds the best model with best hyperparameters, as well as saves the test accuracy score. So in the end, each machine learning algorithm has 10 models and 10 test accuracy scores.

First, I choose a simple logistic regression algorithm with l2 penalty. Its parameter C is tuned, taking values in [0.001, 0.01, 0.1, 1, 10, 100, 1000]. The best logistic regression model with C = 0.1 achieves a test accuracy of 0.88. I also train a random forest algorithm. The max_depth parameter takes values 1, 3, and 10, while the max_feature parameter takes values 0.5, 0.75, and 1. It achieves the best accuracy when max_feature = 0.5 and max_depth = 10. Another machine learning algorithm is the support vector machine classifier (SVC), which has the parameter gamma from 0.01 to 1000 and the parameter C from 0.1 to 10. The best SVC model has a test accuracy of 0.88 when C = 0.1 and gamma = 0.01. Finally, I train XGBoost algorithm and tune the parameters max_depth, learning_rate, as well as the number of estimators. The best XGBoost classifier has parameters max_depth = 5, learning_rate = 0.1, and n_estimators = 150.

To measure the uncertainties of splitting, I look at the variation of test accuracy scores of different machine learning algorithms across different random states, as shown in Figure 4. The error bar plot shows that the uncertainties for different machine learning algorithms are quite similar. The logistic regression algorithm appears to have less variation, i.e. smaller standard deviation, than other algorithms.

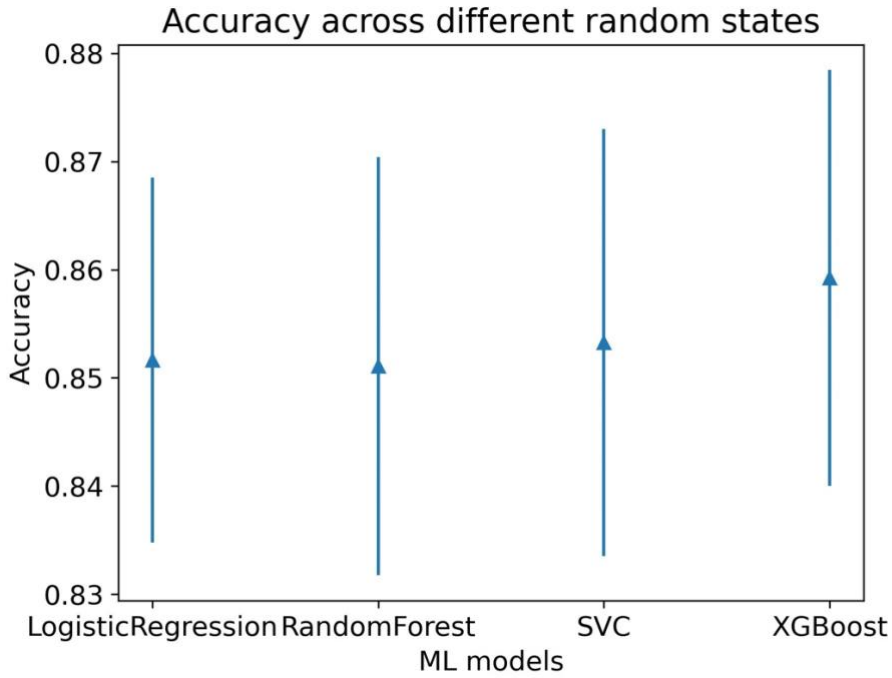

Figure 4. The uncertainties of different machine learning models due to splitting. Error bars represent standard errors.

Results

All machine learning algorithms achieve much higher accuracy than the baseline accuracy, which is 0.55 provided by the very balanced dataset. The accuracy scores of the four algorithms all reach above 85%. As discussed in the methods section, each algorithm is looped through 10 different random states and gets 10 test accuracy scores. Table 1 summarizes the mean and standard deviation of test accuracy scores of each algorithm.

ML_algorithm	Accuracy mean	Accuracy Standard deviation	Number of standard deviations above baseline
Logistic Regression	0.851630	0.016848	17.702790
Random Forest	0.851087	0.019322	15.407699
SVC	0.853261	0.019746	15.187423
XGBoost	0.859239	0.019207	15.924377

Table 1. Results of different machine learning algorithms. The baseline accuracy is 0.55. The number of standard deviations above baseline is calculated by (mean of accuracy – baseline accuracy) / standard deviation of accuracy.

The table above also shows that XGBoost has the highest mean accuracy score and the second-highest number of standard deviations above the baseline. So I decide to choose XGBoost as the final best model. Applying the best XGBoost model to the test set, I get an accuracy of 0.8859 and an F_1 score of 0.8986. The confusion matrix of the model also gives a high precision score of 0.9300 and a high recall score of 0.8692, as shown in Figure 5.

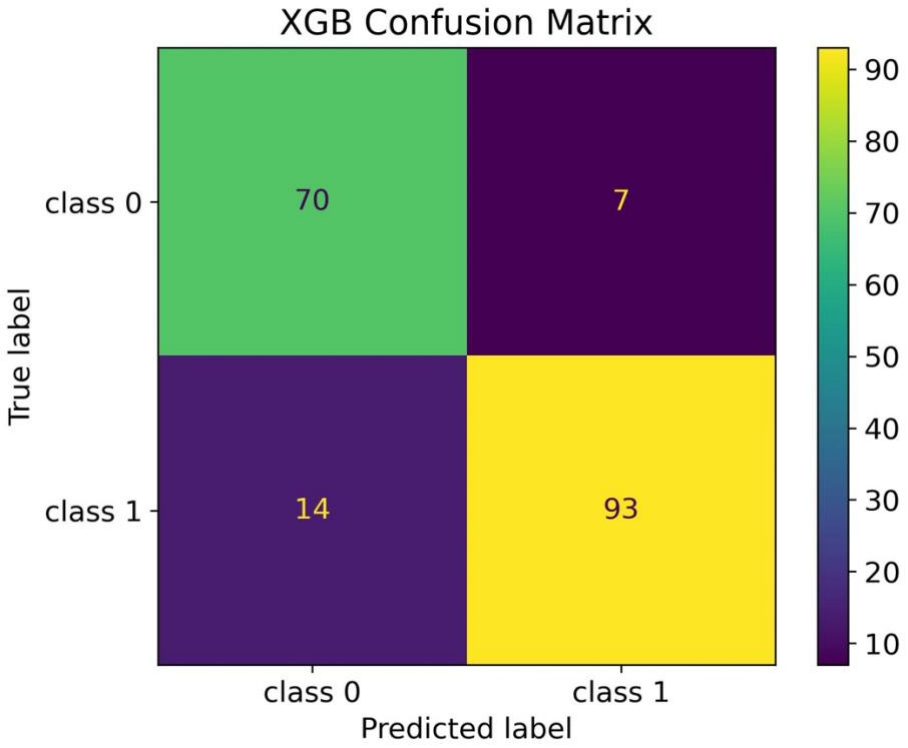

Figure 5. The confusion matrix of the best XGBoost model. Class 1 represents heart disease while class 0 represents no heart disease.

After determining the best model, I use the permutation importance method and 5 different metrics to find important features in predicting heart disease. The six methods give slightly different results when calculating global feature importance, as shown in Figure 6. There are some features that appear to be on the top list regardless of methods: ST slope, Oldpeak, Cholesterol, and Chest Pain Type. Both ST slope and Oldpeak describe the heart rate of a patient, which makes sense in the prediction of heart disease. Cholesterol is a numerical variable and positively affects the possibility of having heart disease. Lastly, the ordinal variable, chest pain type, is also another important feature. It is interesting that age and sex seem to be less important in the prediction while they appear to be correlated to the target variable in the EDA section.

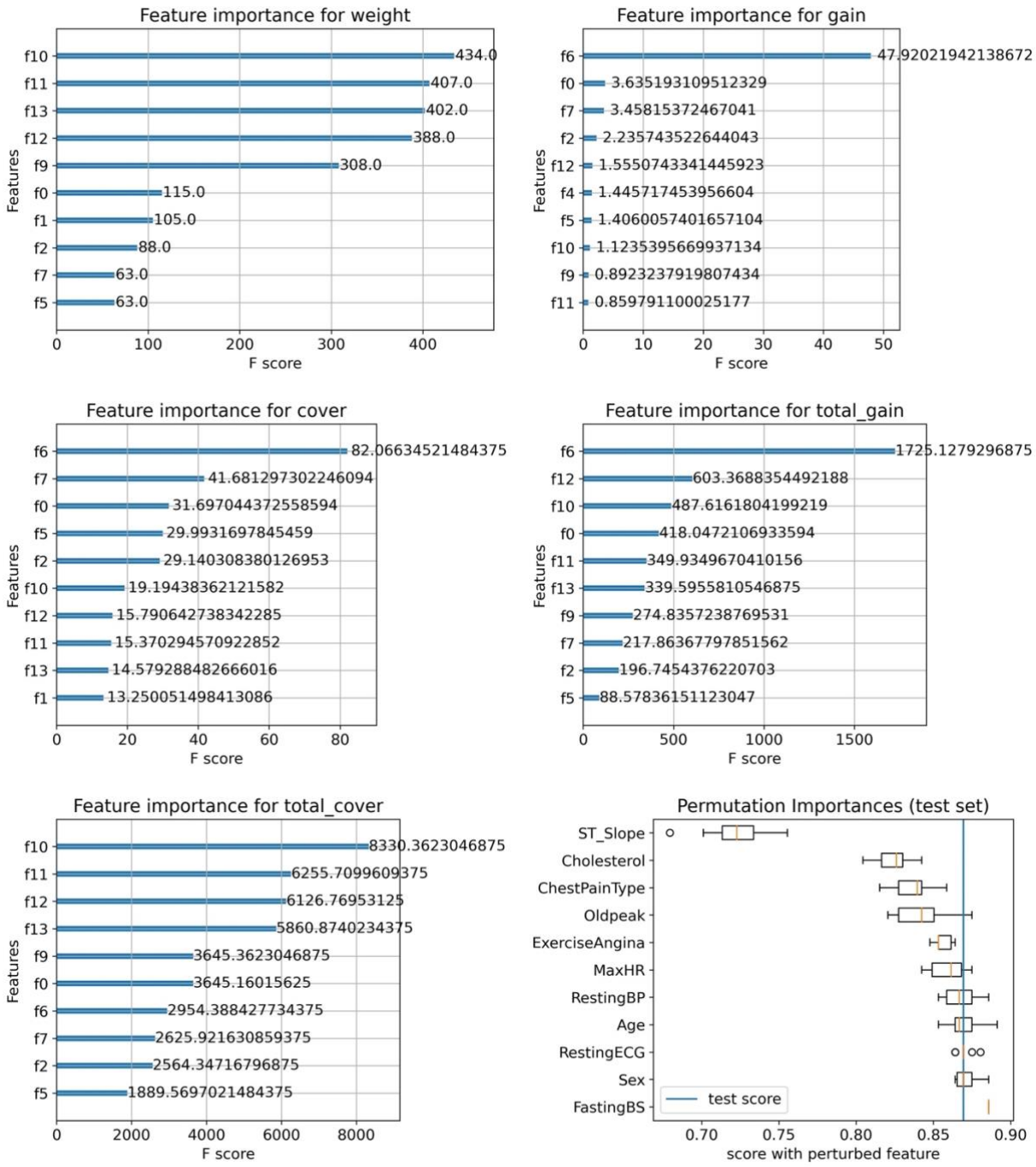

Figure 6. Global feature importance using weight, gain, cover, total_gain, total_cover, and permutation respectively. The f1-f14 features represent ChestPainType, RestingECG, Sex_F, Sex_M, ST_Slope_Down, ST_Slope_Flat, ST_Slope_Up, ExerciseAngina_N, ExerciseAngina_Y, RestingBP, Cholesterol, MaxHR, Oldpeak, and Age respectively.

In addition to global feature importance, local feature importance is also worth noting as it can help medical experts to explain to each patient their own factors affecting their heart disease prediction. I use SHAP values to look into local feature importance for some specific patients. The most important features affecting prediction appear different for each patient, and also different from global important features. For example, the prediction for the 2nd patient in the test set is highly affected by ST slope and RestingECG (resting electrocardiogram results), and is negatively affected by MaxHR (maximum heart rate achieved). Among these important features, only ST slope appears on the top list of global important features.

Figure 7. Local feature importance for the second patient in the test set. The features in red affect the prediction positively while the features in blue negatively affect the prediction.

Outlook

Some further developments could be explored to improve the model's performance. First, while the cholesterol feature of the dataset doesn't contain any missing values, it has some potentially incorrect zero values. The data source doesn't explain well on this feature, so I could find whether these zero values are incorrect or not. In some previous works, people replaced these values with the median value. But the imputation has been proved to introduce bias to the model. People also talk about the possibility that HDL cholesterol and LDL cholesterol cancel each other out, leading to zero values in the cholesterol column. So, I didn't do any imputation in my preprocessing as I tried not to change the dataset. More medical knowledge on cholesterol would help improve the model's performance and interpretability.

In addition, I'd like to decrease false negatives if given more time. The false negatives are higher than false positives, as shown in the normalized confusion matrix (Figure 8). A false negative means that the model fails to recognize a patient having heart disease. So, lowering false negatives would help medical experts to identify as many heart diseases as possible, thus helping more patients get effective treatments.

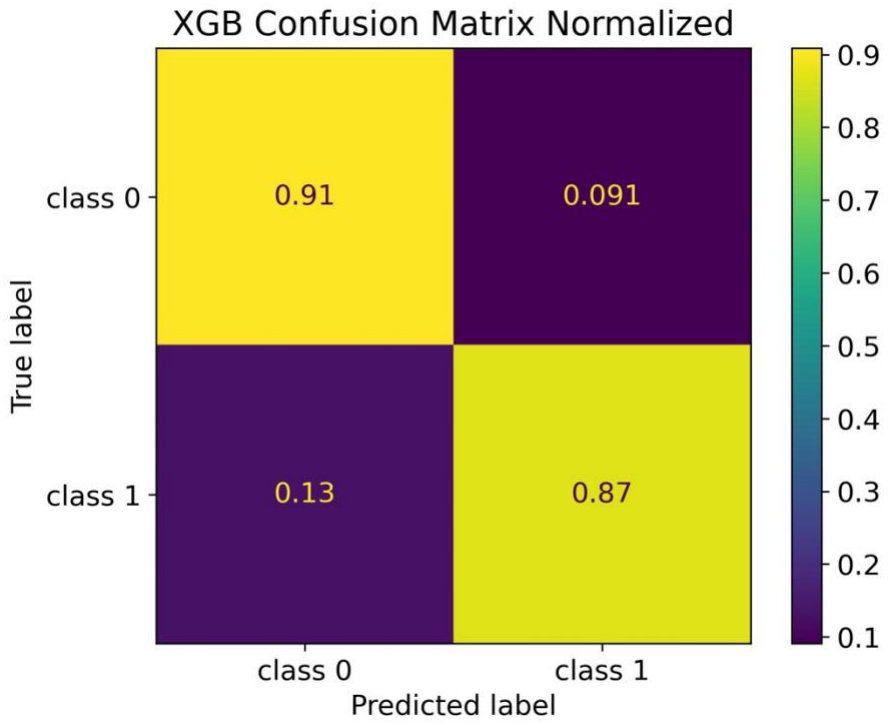

Figure 8. The normalized confusion matrix for the best XGBoost model. The original confusion matrix is normalized with respect to the true conditions. It shows relatively high false negatives.

References

Centers for Disease Control and Prevention. (2022, June 3). Target heart rate and estimated maximum heart rate. Centers for Disease Control and Prevention. Retrieved October 17, 2022, from https://www.cdc.gov/physicalactivity/basics/measuring/heartrate.htm

Centers for Disease Control and Prevention. (2022, October 14). Heart disease facts. Centers for Disease Control and Prevention. Retrieved October 17, 2022, from https://www.cdc.gov/heartdisease/facts.htm#:~:text=One%20person%20dies%20every%2034,United%20States%20from%20cardiovascular%20disease.&text=About%20697%2C000%20people%20in%20the,1%20in%20every%205%20deaths.

fedesoriano. (September 2021). Heart Failure Prediction Dataset. Retrieved [Date Retrieved] from https://www.kaggle.com/fedesoriano/heartfailure-prediction.

Patel, J., TejalUpadhyay, & Patel, S. (2016). Heart Disease Prediction Using Machine learning and Data Mining Technique. https://doi.org/10.090592/IJCSC.2016.018

World Health Organization. (n.d.). Cardiovascular diseases. World Health Organization. Retrieved October 17, 2022, from https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1

Github Repository

https://github.com/anran2176/Heart_Disease_Prediction