Heart Health Inspector

Ahmed Aladdin Mohammed

Cairo University

Machine Learning Frameworks

Ahmed Hany, Basma Hatem, Rofayda Bassem

2024/8

1- Exploratory Data Analysis (EDA)

About the Dataset

The provided dataset includes the medical information about healthy and others with heart dieases. It include data about their age, sex, the type of pain they have in the chest, their resting blood pressure, cholesteros reading if measured otherwise it is set to zero, fasting blood sugar, resting electrocardiography, maximum heart rate, whether they have an exercise angina or not, oldpeak, ST slope, and finally whether that person has a heartdisease or not.

Exploring the data

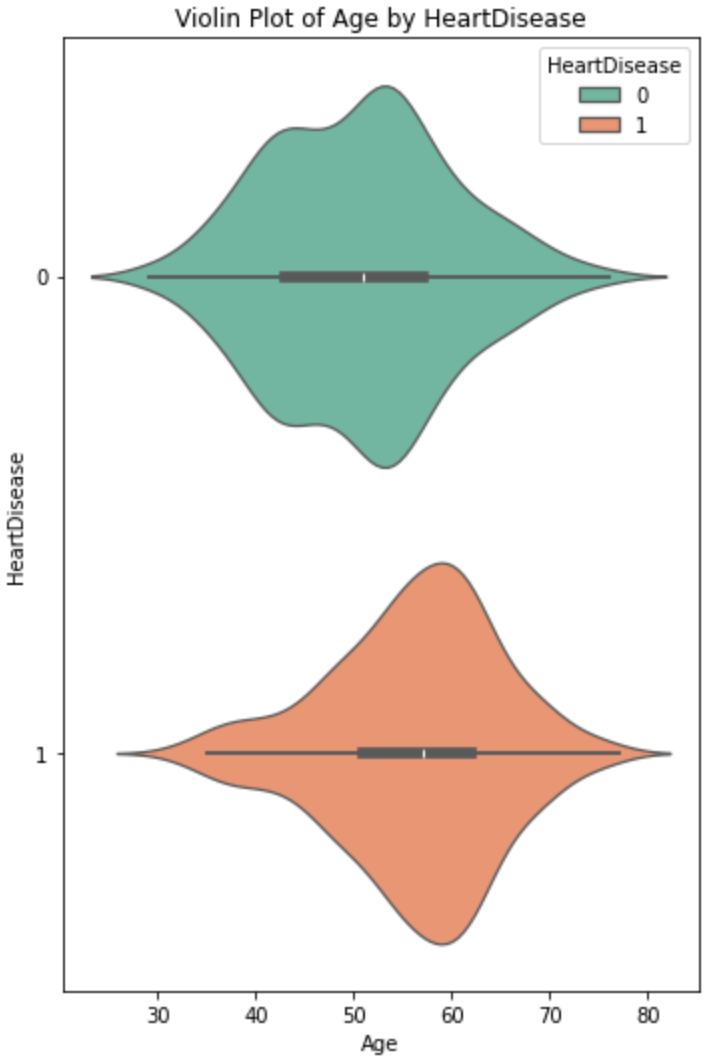
We started with defining the target in this dataset. This dataset was meant to be using in training a model to detect heart disease based on the given medical information, so the target was the ‘HeartDisease’ column in the dataset

I started exploring the data by showing the datatype of each feature and the number of null values in the column of each feature. Luckily, no null values were found in the data.

Then, the set of features were divided into a set of categorical and numerical features, with also taking some of the numerical features that had some certain values - no a range - and added them into the categorical features set.

For the categorical features, I identified the different categories of each individual feature by graphical means (count plot), which also helped in identifying the categories distributions. Also, using a bar plot, I was able to visualize the distribution of contribution of each category in having a heart disease or not. Finally, the Chi-Square score was used to measure the strength of the relation between each categorical feature and the target.

For the numerical features, I started by displaying some info about each feature like the mean, standard deviation, min, and max, then using a hist plot, we could visualize the distribution of the data, and we can note than all the numerical features had a tendency to normal distribution. Then using a violin plot, we could visualize the spread and distribution of the numeric variable across having and not having a heart disease. Finally I showed the relation between the the these features and each other and with the target using the correlation matrix.



2- Preprocessing

Remove Insignificant Features

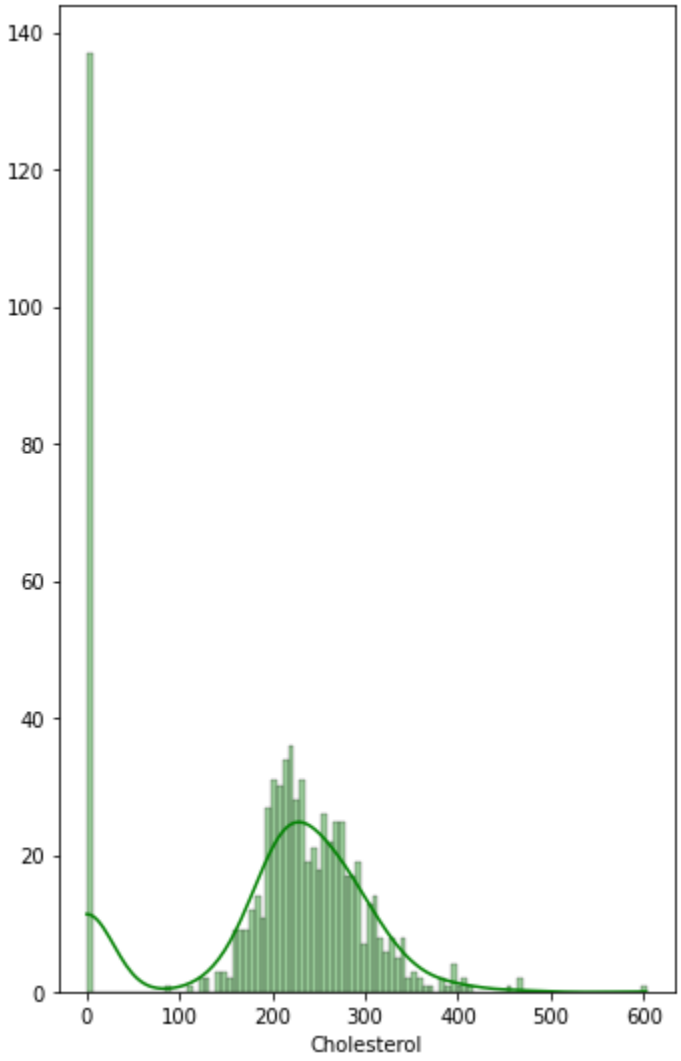
Here, I tried removing the features that showed week contribution to the target, whether in the Chi-Square test or in the correlation matrix, but that only lead to a worse prediction results for the model.

Remove Outliers

In the EDA stage, we could see some outliers in some features (RestingBP, Cholesterol, Oldpeak), and the number of these outliers were so small that it can be removed safely from the dataset, which I did.

Deal with the Large Number of Zeros

Some features (Cholesterol, Oldpeak) showed a very large number of zeros, which in some cases where because of missing measurements like in the case of Cholesterol - no human would have a cholesterol level of zero - and the number of these zeros we large that the best solution in this case was to add new features for each of these two (Zero_Oldpeak, Have_Cholesterol_Measurement) to indicate these features were zeros or not; on the hope that the behavior of the model would change in case of having zero and not having zero.



Encode the Categorical Features

For binary categorical features like Sex and ExerciseAngina, I maped there values to either ONE or ZERO. For the other categorical features, One-Hot encoding was used.

Standardize the Numerical Features

For the Numerical Features, and due to having different ranges which could lead to some features having greater wights than others, I decided to go with using the Standard Scaler. **Note:** before applying the standard scalar, I divided the dataset into training and validation datasets to prevent data leakage of the validation set.

3- Model Selection and Evaluation Criteria

For this project, I went with experimenting three different models: KNN, Logistic Regression, and Forward Neural Networks.

My selection and evaluation criteria did depend on the accuracy of each model and the stability in getting that accuracy, and in the end I settled on choosing the KNN model.

4- Explanation of Work

Data Section

First, I started with exploring the dataset I was going to work with, identify the different patterns, distributions, and anomalies/outliers in the data, and the contribution of each feature to the target.

Second, in the dataprocessing stage, I removed the outliers, used feature engineering to deal with the large number of zeros in some features; separate the data into train and validation sets; encoded the categorical features and standardize the numerical ones.

Models Section

Third, the models building stage, where I started by building a KNN model and used Optuna to test a set of different hyperparameters (n_neighbores, metric) and then build the real model based on the best hyperparameters combination.

For the second model, I went with Logistic Regression with which I also used Optuna to choose the best combination of number of epochs and batch size.

For the Forward Nural Network model, I tried manually testing different combinations of the number of layers and and number of neurons of those layers.

Selecting the best model was as follows, the KNN model gave the stable accuracy of 86% which surprisingly was the best among the three of them; the Logistic Regression model gave different results based on the initial weights it started with, and the best accuracy it got was about 80%; for the FNN model, it also had different outcomes based on the initial value of the wights and the number of layers and neurons, and most of the time it gave results between the other two models and rarely above the KNN model; at the end, I decided to stick with the KNN model having a decent and stable accuracy.

Customer Application

I created a good looking web page that provides a simple user experience for individuals to be able to enter their medical info and get an inspection of whether they might have a Heart Disease or not. The app was deployed using Docker and Azura to enable access from around the globe.

5- Comparison between Approaches

KNN

The KNN model was the most intuitive and simple solution for this problem, and although it can be computationally expensive due to the approach it follows, we could look down on that given that we have a small dataset; which would also help in the prediciton time.

Logistic Regression (Machine Learning Approach)

Although the LR is alos simple and is computationally efficient, but due to the non-linearity between the features and target, it didn't perform well.

Forward Neural Network (Deep Learning Approach)

The FNN is a more complex approach that requires longer training times and it requires more resources. Due to the small dataset, the FNN model suffered from overfitting in which it easily reached above 90% accuracy on the training set but when it was tested on the validation set it gave a maximum of 86% accuracy and many times were in range of 84%. And in our case that was a real problem that couldn't be helped.