Heart Disease Prediction App - Capstone Proposal

Introduction

Cardiovascular diseases (CVDs) is a disorder group that affects the heart and blood vessels that some of them may result in heart attacks and strokes that are acute events and very letal.

According to the World Health Organization, more than 17 million people pass away every year by CVDs, being the leading cause at 31,59% of all deaths. And the most common CVDs is ischaemic heart disease, representing 16% of all losses.

Acting early at the first signs of a possible disease is always the best alternative for avoiding being ill, and that's why scientists and doctors are always researching for signals that may point out possible risks and start treatments as soon as possible.
But would it be possible to estimate risks within patient blood and heart data?

Problem statement

Considering a dataset containing some patients data around the world, which has some blood and heart profile, what are the variables that are significant enough to estimate risks of a patient being?
And is it possible to build an accurate machine learning model capable of giving prognostics based on people's data?

Dataset and Inputs

The project and the model will be based on a dataset found on kaggle, with contains samples of patients presenting the following informations:

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]

Below is the source of the dataset:
https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/

Solution

As mentioned in the problem statement, the project's goal is to construct a classification machine learning model that can collect a given patient data and return an answer telling if the patient has or not the risk of having a CVD.

In order to make it user-friendly, the built model will be imported and deployed on a simple web application using the Streamlit framework.
The idea is to build a simple online form where users can input their own data and receive the model's output.

Benchmark Model

The benchmark model was selected based on a research with 'Heart Disease Prediction' keywords and the Classification approach and methodology, and resulted in the following article:

https://www.analyticsvidhya.com/blog/2022/02/heart-disease-prediction-using-machine-learning/#h2_4

Aman's article has a similar dataset, with some different variables, but the objective was to find out the most significant ones and build a model based on them.
His case focused only on exploring the KNN Algorithm and evaluated using just the Accuracy of the model.

So, besides testing KNN, this project also will explore other algorithms and evaluation metrics as a way to compare to Aman's model.

Set of evaluation Metrics

A python module called PyCaret offers an automated way to explore models and iterate over several of them, making it easier to focus on adjusting parameters. It also provides a set of evaluation metrics for classification problems, such as:

- Accuracy
- Precision
- Recall
- AUC
- F1
- Kappa
- MCC

But there’s not much sense in using all of them. A clever approach is to consider the problem’s domain before selecting one.

As the solution works around a health care solution, the smarter decision is to use Recall as the evaluation metric, hence its priority to focus on watching the amount of False-Negatives.

So, the best model will be the one that has the best Recall.

Project Design

The final objective of the project is to deploy an online web application using Streamlit, where anyone could simply input their data and receive a prognostic about their risk of having a CVD.

In order to build it, the necessary steps are the following:

- Collect the dataset from Kaggle
    - Download data
    - Load on notebook
- Data cleaning and preprocessing
    - Removing null data
    - Formatting numbers and strings
- Exploratory Data Analysis
    - Explore features
    - Check distribution of the variables
    - Balance the dataset (closes to 50/50 targets)
    - Pearson’s correlation heatmap
    - Removing outliers
    - Removing non-sense data
- Setup Pycaret
    - Run initialize function
    - Collect insight about the output
    - Run the first sample with a 0.8 split ratio (train/test)
- Selection of the model
    - Run the Compare Models - Pycaret function
    - Select the best algorithms models based on the evaluation metric (Recall)
    - Boost model using Recall as the goal
- Model evaluation
    - Run the evaluation function
    - Explore the confusion Matrix
    - Import to pickle the best model
- Streamlit forms
    - Build a simple forms with the input data fields
    - Create a call function
    - Load the model
    - Test and deploy the app