**Heart disease prediction using different machine learning methods**

# **Project description**

Heart disease (CVD) is common diseases that cost hundreds of deaths all over the world. Its hard for the doctors to predicate whether the patient could get a heart disease or not that could save his life and give him an early treatment. Thus, by using machine learning we can predicate that someone could get a heart disease or not. In this project different machine learning methods were used with some techniques and compared with published paper to get the best results this data could produce.

## **The machine learning models that were used:**

* Logistic Regression
* Naïve Bayes
* K-Nearest neighbours
* Decision Tree

## **Machine learning techniques that were used for the models:**

* SMOTE
* Data split using cross validation K-Fold = 10
* Grid search
* Feature selection

**Libraries used:**

* **Mostly Sklearn libaraies for models, classification report, metrices**
* **Matplotlib**
* **Pandas**
* **numpy**

# **Data Descriptions**

1. **age:** displays the age of the individual.
2. **sex:** displays the gender of the individual using the following format :
3. 1 = male
4. 0 = female
5. **cp (Chest-Pain Type):** displays the type of chest-pain experienced by the individual using the following format:
   1. 0 = typical angina
   2. 1 = atypical angina
   3. 2= non — anginal pain
   4. 3 = asymptotic
6. **trestbps(Resting Blood Pressure):** displays the resting blood pressure value of an individual in mmHg (unit)
7. **chol(Serum Cholestrol):** displays the serum cholesterol in mg/dl (unit)
8. **fbs (Fasting Blood Sugar):** compares an individual's fasting blood sugar value with 120mg/dl. If fasting blood sugar > 120mg/dl
   1. 1 (true) else
   2. 0 (false)
9. **restecg (Resting ECG):** displays resting electrocardiographic results • 0 = normal
   1. 1 = having ST-T wave abnormality
   2. 2 = left ventricular hyperthrophy
10. **thalach(Max Heart Rate Achieved):** displays the max heart rate achieved by an individual.
11. **exang (Exercise induced angina):**
    1. 1 = yes
    2. 0 = no
12. **oldpeak (ST depression induced by exercise relative to rest):** displays the value of an integer or float.
13. **slope (Peak exercise ST segment):**
    1. 0 = upsloping,
    2. 1 = flat
    3. 2 = downsloping
14. **ca (Number of major vessels (0–3) colored by fluoroscopy**): displays the value as integer or float.
15. .**thal: displays the thalassemia (is an inherited blood disorder that causes your body to have less hemoglobin than normal)** :
    1. 0 = normal
    2. 1 = fixed defect
    3. 2 = reversible defect
16. **target (Diagnosis of heart disease):** Displays whether the individual is suffering from heart disease or not:
    1. 0 = absence
    2. 1 = present

# **Documentation**

* We started by reading the csv file and knowing the meaning of each variable whether continues or discrete. Then, visualizing the dataset we had 14 columns, 13 features and 1 target.
* Data was clean of nulls.

## **EDA:**

* We visualized the data using head ()
* Then we done EDA by pp.ProfileReport(df) which produced a report the amount of male and female to the target and reported many data visualization.

## **Outlier Detection and removal**

* Then we did outlier detection using IQR which we find the upper and lower limit and any then that does not lie between these two boundaries get removed by converting them to null then remove the null values.
* We repeated some of outlier detection function as some of the outliers did not remove so, we had to repeat it.

## **Data split**

* After making sure the data was clean, we then detect the categorical values and the continuous and convert the categorical to dummy variables to see their influence on the target.
* Then we split the data to X and Y using split function from sklearn matrices 25 test data and 75 train data. Then we used StandardScaler() to standardize the data points.
* For naïve bayes we took copy of the data without the dummy variables and split for X and Y different.

## **Logistic regression**

* We applied the first model Logistic regression with the default parameters. Then we used cross validation to evaluate the model accuracy as by running every time the accuracy change.
* Then we applied LR as the paper to compare the results. The paper used SMOTE which an over sampling method by K- nearest neighbors and split the data using K-fold cross validation =10 folds. We applied those techniques to compare the result.
* We made a pipeline for the paper that start with smote, stander scalar then finally the model.
* The result nearly the same as the paper
* Then we used the same techniques and applied grid search to see if there going to be any improve the model for better results. By tunning the hyperparameter C.
* We applied the same pipeline of the paper, but we added the grid search. So, that the grid search function takes the pipeline and the different parameters and find the best one with the highest accuracy.
* Lastly, we applied pipeline Logistic regression using Recursive Feature Elimination on dataset features, smote (random\_state=11)and grid search for the features. we applied this 2 times, one with dummy variables and the other the dataset as it is.

## **Naïve bayes**

* Then we applied naïve bayes to the data one with the correlated columns and another without the correlated columns then we applied the same techniques of the paper. The paper did not have naïve bayes we applied as an extra to compare different models.
* We applied the pipeline with SMOTE with the 2 data the (one with the correlated columns and one without) then we applied the pipeline of the grid search by tunning the Laplace smoothing parameter.

## **KNN**

* We applied KNN model with a function that search for the best K and result the accuracy then we compared with the smote pipeline of the paper with K=2.
* We applied the grid search pipeline to KNN and result the best accuracy the same as the first KNN model (without the smote).

## **Decision tree**

* We finally applied Decision tree model with the default parameters, then with smote pipeline then with grid search pipeline by tunning the maximum depth.

## **Evaluation of models**

* Finally, we compared the accuracy of each method using **y test** and y predicate and with ROC by the FP and TP rate.
* We visualized the feature importance of each model except for KNN (as mentioned in the paper). By using permutation\_importance