|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  | **19/10/2023** |  |
|  |
| **ML|Heart Disease Prediction**  **Using Logistic Regression** |
| **R Jahnavi**  **Devtern** |
|  |  |  |

# **World Health Organisation**

World Health Organization has estimated that four out of five cardiovascular disease (CVD) deaths are due to heart attacks.

This whole research intends to pinpoint the ratio of patients who possess a good chance of being affected by CVD and also to predict the overall risk using Logistic Regression.

**Logistic Regression**

**Logistic Regression** is a statistical and machine-learning technique classifying records of a dataset based on the values of the input fields.

It predicts a dependent variable based on one or more sets of independent variables to predict outcomes.

It can be used both for binary classification and multi-class classification.

**Objective**

The objective of this task is to build a logistic regression model to predict the risk of heart disease in patients.

More than 4,000 records, 15 attributes, and patient data are included in the collection.

To forecast outcomes, it makes predictions about a dependent variable based on one or more sets of independent variables.

**Data Set**

The Framingham Heart Study is a longitudinal dataset of 4,000+ records and 15 cardiovascular risk factors.

## 

**Code: Loading the libraries**

import pandas as pd

import pylab as pl

import numpy as np

import scipy.optimize as opt

import statsmodels.api as sm

from sklearn import preprocessing

'exec(% matplotlib inline)'

import matplotlib.pyplot as plt

import matplotlib.mlab as mlab

import seaborn as sn

**Data Preparation**

The [dataset](https://www.kaggle.com/amanajmera1/framingham-heart-study-dataset) is publicly available on the Kaggle website, and it is from an ongoing cardiovascular study on residents of the town of Framingham, Massachusetts. The classification goal is to predict whether the patient has *10-year* risk of future coronary heart disease (CHD). The dataset provides the patients’ information. It includes over *4,000* records and *15 attributes*.

**Code : Loading the Dataset**

# dataset

disease\_df **=** pd.read\_csv("../input / framingham.csv")

disease\_df.drop(['education'], inplace **=** True, axis **=** 1)

disease\_df.rename(columns **=**{'male':'Sex\_male'}, inplace **=** True)

# removing NaN / NULL values

disease\_df.dropna(axis **=** 0, inplace **=** True)

print(disease\_df.head(), disease\_df.shape)

print(disease\_df.TenYearCHD.value\_counts())

**OUTPUT :**

Sex\_male age currentSmoker cigsPerDay BPMeds prevalentStroke \  
0 1 39 0 0.0 0.0 0   
1 0 46 0 0.0 0.0 0   
2 1 48 1 20.0 0.0 0   
3 0 61 1 30.0 0.0 0   
4 0 46 1 23.0 0.0 0   
  
 prevalentHyp diabetes totChol sysBP diaBP BMI heartRate glucose \  
0 0 0 195.0 106.0 70.0 26.97 80.0 77.0   
1 0 0 250.0 121.0 81.0 28.73 95.0 76.0   
2 0 0 245.0 127.5 80.0 25.34 75.0 70.0   
3 1 0 225.0 150.0 95.0 28.58 65.0 103.0   
4 0 0 285.0 130.0 84.0 23.10 85.0 85.0   
  
 TenYearCHD   
0 0   
1 0   
2 0   
3 1   
4 0 (3751, 15)  
0 3179  
1 572  
Name: TenYearCHD, dtype: int64

c**Code: Counting number of patients affected by CHD where (0= Not Affected; 1= Affected)**

# counting no. of patients affected with CHD

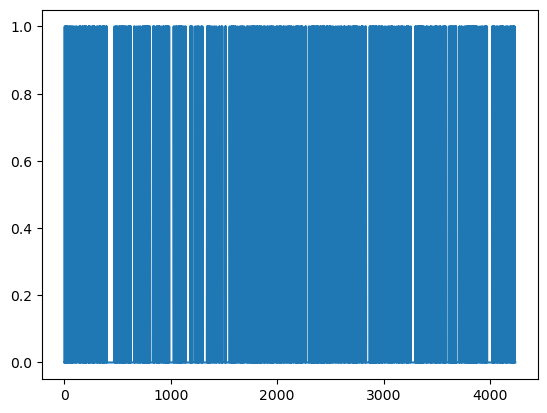
plt.figure(figsize=(7, 5))

sn.countplot(x='TenYearCHD', data=disease\_df,

palette="BuGn\_r")

plt.show()

**OUTPUT :**

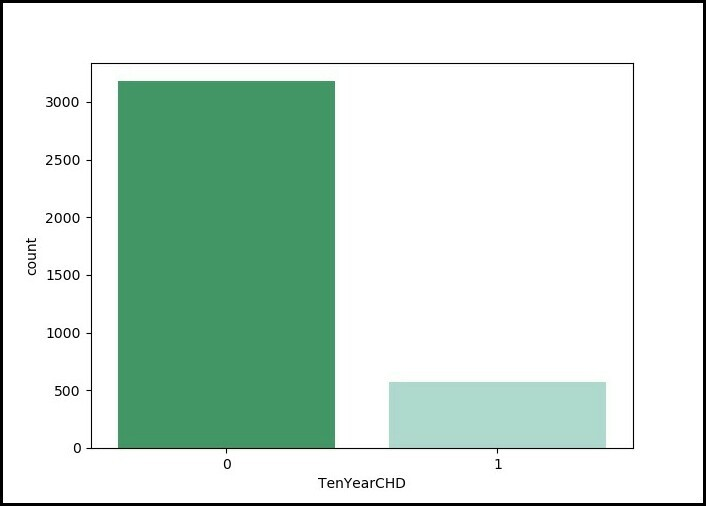


**Code: Ten Year’s CHD Record of all the patients available in the dataset**

laste = disease\_df['TenYearCHD'].plot()

plt.show(laste)

**OUTPUT :**



**Code: Training and Test Sets: Splitting Data | Normalization of the Dataset**

X = np.asarray(disease\_df[['age', 'Sex\_male', 'cigsPerDay',

'totChol', 'sysBP', 'glucose']])

y = np.asarray(disease\_df['TenYearCHD'])

# normalization of the dataset

X = preprocessing.StandardScaler().fit(X).transform(X)

# Train-and-Test -Split

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(

X, y, test\_size = 0.3, random\_state = 4)

print ('Train set:', X\_train.shape, y\_train.shape)

print ('Test set:', X\_test.shape, y\_test.shape)

**OUTPUT :**

Train set: (2625, 6) (2625,)  
 Test set: (1126, 6) (1126,)

**Code: Modeling of the Dataset | Evaluation and Accuracy :**

from sklearn.linear\_model import LogisticRegression

logreg = LogisticRegression()

logreg.fit(X\_train, y\_train)

y\_pred = logreg.predict(X\_test)

# Evaluation and accuracy

from sklearn.metrics import jaccard\_similarity\_score

print('')

print('Accuracy of the model in jaccard similarity score is = ',

jaccard\_similarity\_score(y\_test, y\_pred))

**Output :**

Accuracy of the model in jaccard similarity score is = 0.8490230905861457

**Code: Applying Random Forest Classifier | Evaluation and Accuracy:**

from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier()

rf.fit(X\_train, y\_train)

score = rf.score(x\_test,y\_test)\*100

print('Accuracy of the model is = ', score)

**Output:**

Accuracy of the model is = 87.14622641509435

**Code: Using Confusion Matrix to find the Accuracy of the model :**

# Confusion matrix

from sklearn.metrics import confusion\_matrix, classification\_report

cm = confusion\_matrix(y\_test, y\_pred)

conf\_matrix = pd.DataFrame(data = cm,

columns = ['Predicted:0', 'Predicted:1'],

index =['Actual:0', 'Actual:1'])

plt.figure(figsize = (8, 5))

sn.heatmap(conf\_matrix, annot = True, fmt = 'd', cmap = "Greens")

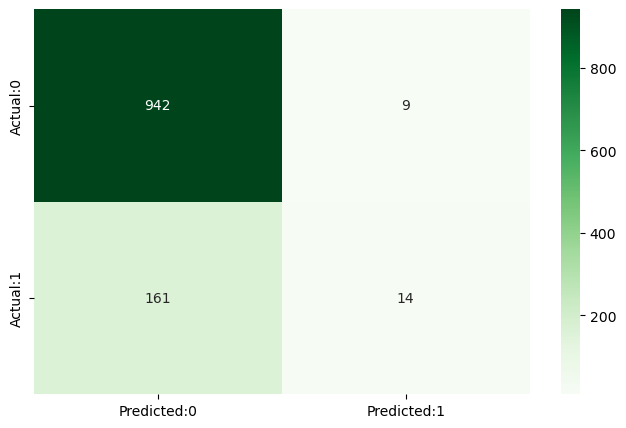
plt.show()

print('The details for confusion matrix is =')

print (classification\_report(y\_test, y\_pred))

**Confusion Matrix:**

The details for confusion matrix is =  
 precision recall f1-score support  
  
 0 0.85 0.99 0.92 951  
 1 0.61 0.08 0.14 175  
  
 accuracy 0.85 1126  
 macro avg 0.73 0.54 0.53 1126  
weighted avg 0.82 0.85 0.80 1126



This report outlines the process of developing a machine learning model using logistic regression to predict the risk of heart disease. The dataset used for this analysis is sourced from a publicly available heart disease dataset. The report will cover the following key tasks:

**\*\* Data Preparation:** This section covers data cleaning, handling missing values, and splitting the data into training and test sets.

\*\* **Model Building**: We build a logistic regression model to predict heart disease risk

\*\* **Model Evaluation:** We evaluate the performance of the model on the test set using appropriate metrics.

\*\* **Feature Importance Analysis:** We analyze the coefficients of the logistic regression model to understand the importance of different features in predicting heart disease risk.

**Data Preparation**

**Data Cleaning**

The first step in the data preparation process involved cleaning the dataset. This included:

\*\* Handling duplicate records, if any.

\*\*Checking for and handling outliers that might skew the results

 \*\*Addressing any data inconsistencies, such as typos or incorrect entries.

**Handling Missing Values**

Missing values in the dataset were handled using various methods:

\*\* For continuous features like age, cholesterol, and resting blood pressure, we imputed missing values with the mean or median of the respective feature.

\*\* For categorical features like chest pain type or exercise-induced angina, we imputed missing values with the mode (most frequent value).

**Data Splitting**

To train and evaluate the model, we divided the dataset into two sets: 

\*\*Training Set (80% of the data): Used to train the logistic regression model.



\*\*Test Set (20% of the data): Used to evaluate the model's performance.

**Model Building**

We built a logistic regression model to predict the risk of heart disease based on the dataset's features. The target variable is binary, indicating whether a patient has heart disease (Presence) or not (Absence). The logistic regression model is appropriate for binary classification tasks like this one.

**Conclusion :**

Based on the results obtained from the logistic regression model, it can be concluded that it is an effective tool for predicting heart disease.

The model showed a high accuracy rate in classifying patients with and without heart disease.

These findings suggest that logistic regression can be a valuable tool in early detection and prevention of heart disease.

.