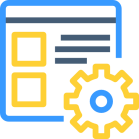
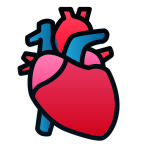
|  |
| --- |
|  |
| HEART-DISEASE Prediction model |
| Built using: MACHINE LEARNING , Logistic Regression Model, Python , Pandas, Numpy, Seaborn, Matplotlib, Scikit-learn libraries |



OBJECTIVE: To analyze a given dataset consisting of health information of variuos patients and devising a responsive MACHINE-LEARNING model to prescribe whether the person is suffering from a heart-related disease or not. The analysis is carried out on a Jupyter notebook.

DESCRIPTION ABOUT THE DATASET:

1**. Age (age): Patient's age in years.**

**2. Sex (sex): Gender of the patient.**

**Values: 1 = Male, 0 = Female**

**3. Chest Pain Type (cp): Type of chest pain experienced.**

**Values: 1 = Typical angina, 2 = Atypical angina, 3 = Non-anginal pain, 4 = Asymptomatic**

**4. Resting Blood Pressure (trestbps): Blood pressure on admission in mm Hg.**

**5. Serum Cholesterol (chol): Serum cholesterol level in mg/dl.**

**6. Fasting Blood Sugar (fbs): Fasting blood sugar level.**

**Values: 1 = >120 mg/dl, 0 = <=120 mg/dl**

**7. Resting Electrocardiographic Results (restecg): Results of resting electrocardiogram.**

**Values: 0 = Normal, 1 = ST-T wave abnormality, 2 = Probable or definite left ventricular hypertrophy**

**8. Maximum Heart Rate Achieved (thalach): Maximum heart rate during examination.**

9. **Exercise-Induced Angina (exang): Presence of exercise-induced angina.**

**Values: 1 = Yes, 0 = No**

**10. ST Depression Induced by Exercise Relative to Rest (oldpeak): ST depression induced by exercise relative to rest.**

**11. Slope of the Peak Exercise ST Segment (slope): Slope of the peak exercise ST segment.**

**Values: 1 = Upsloping, 2 = Flat, 3 = Downsloping**

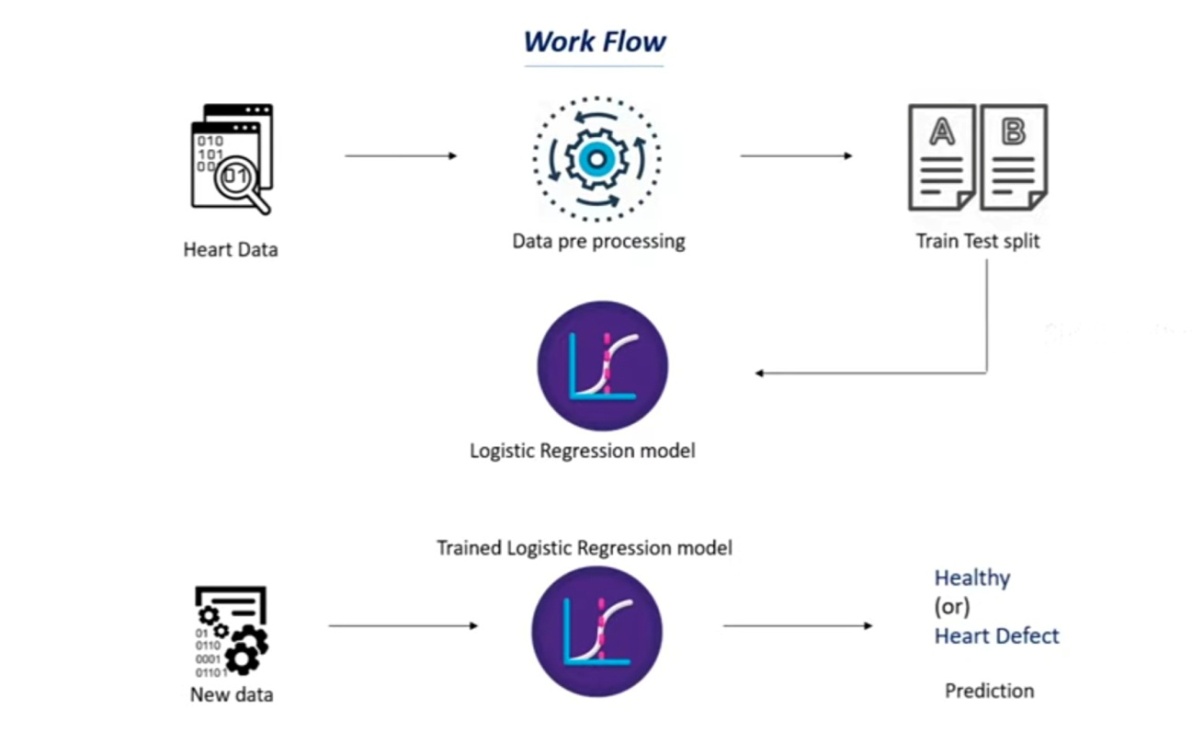
**12. Number of Major Vessels Colored by Fluoroscopy (ca): Number of major vessels colored by fluoroscopy. A higher count may indicate a greater degree of vessel involvement or narrowing, which can be associated with more advanced stages of coronary artery disease.**

**13. Thalassemia (thal): Type of thalassemia.**

**Values: 3 = Normal, 6 = Fixed defect, 7 = Reversible defect**

**14.Target:    Values: 0 = Healthy, 1 = Has a heart-related disease)**

Workflow of the program ->



STEPS:

STEP-1: Collecting and filtering required dataset from resources providing information about patients health categories and combining it all into an Excel worksheet.

STEP-2: Opening a jupyter notebook and importing the depencies(including the libraries Pandas, Numpy, Seaborn, Matplotlib and Scikit-learn).

STEP-3: Importing the dataset which is in the form of csv file.

STEP-4: Data collection and preprocessing to identify any null values and gather basic idea about the data.

STEP-5: Checking the distribution of target variable where-

1 -> DEFECTIVE Heart

0 -> HEALTHY Heart

STEP-6: Splitting ‘features’ and ‘target’:

X = heart\_data.drop(columns = 'target', axis = 1)

Y = heart\_data['target']

print(X)

|  |
| --- |
|  |

STEP-7: Choose test data size to be 20% of the dataset:

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X,Y, test\_size = 0.2, stratify=Y,random\_state= 2)

print (X.shape, X\_train.shape, X\_test.shape)

STEP-8: Model Testing using Logistic regression and finding the accuarcy on respective type of train/test data:

model= LogisticRegression()

model.fit(X\_train,Y\_train

X\_train\_prediction = model.predict(X\_train)

training\_data\_accuracy =accuracy\_score(X\_train\_prediction,Y\_train

print('Accuracy on training data is' ,training\_data\_accuracy )

X\_test\_prediction = model.predict(X\_test)

testing\_data\_accuracy =accuracy\_score(X\_test\_prediction,Y\_test)

print('Accuracy on testing data' ,testing\_data\_accuracy )

STEP-9: Building a predictive system of evaluation:STEP-10

input\_data = (41,0,1,130,204,0,0,172,0,1.4,2,0,2)

#changing the inut data as numpy array

input\_data\_as\_numpy\_array = np.asarray(input\_data)

#reshaping the numpy array as we are predicting for only an instance

input\_data\_reshaped = input\_data\_as\_numpy\_array.reshape(1,-1)

prediction = model.predict(input\_data\_reshaped)

print (prediction)

if ( prediction[0]== 0):

   print ('The person does not have a Heart Disease.')

else:

   print ('The person is suffering from a Heart Disease.’)

SYUGGFFYYGYGYYG6YUGYGYGYYHHBGYGYGVGGVGGGGGTVYGVYUUUSTEP-9: Building a predictive systemAccuracy on tr Summary keypoints:

* Total number of **columns** in the dataset are **14** and **rows** are **303.**
* Accuracy on the **TRAINING Data** was obtained as **85%**, on **TEST Data** as **82%**.
* With the given dataset **no** issue of **overfitting** or **underfitting** was found.
* Extremely vital in present times as **early detection** helps in preventive care and reduced mortaltiy rates.

aining data is 0.8512396694214877

Ssummarey S  
 Accuracy on training data is 0.8512396694214877

ccuracy on training data is 0.8512396694214877

Accuracy on training data is 0.8512396694214877