**Heart Disease Detection with Machine Learning**

## Introduction

Machine learning (ML) is a branch of artificial intelligence (AI) that is increasingly utilized within the field of cardiovascular medicine. It is essentially how computers make sense of data and decide or classify a task with or without human supervision. Machine learning can accurately predict cardiovascular disease and guide treatment—but models that incorporate social determinants of health better capture risk and outcomes for diverse groups. People with cardiovascular disease or who are at high cardiovascular risk (due to the presence of one or more risk factors such as hypertension, diabetes, hyperlipidaemia or already established disease) need early detection and management wherein a machine learning model can be of great help

## Significance

Machine learning—a type of artificial intelligence used to detect patterns in data—is being rapidly developed in cardiovascular research and care to predict disease risk, incidence, and outcomes. Already, statistical methods are central in assessing cardiovascular disease risk and U.S. prevention guidelines. Developing predictive models gives health professionals actionable information by quantifying a patient’s risk and guiding the prescription of drugs or other preventive measures.

## Dataset Description

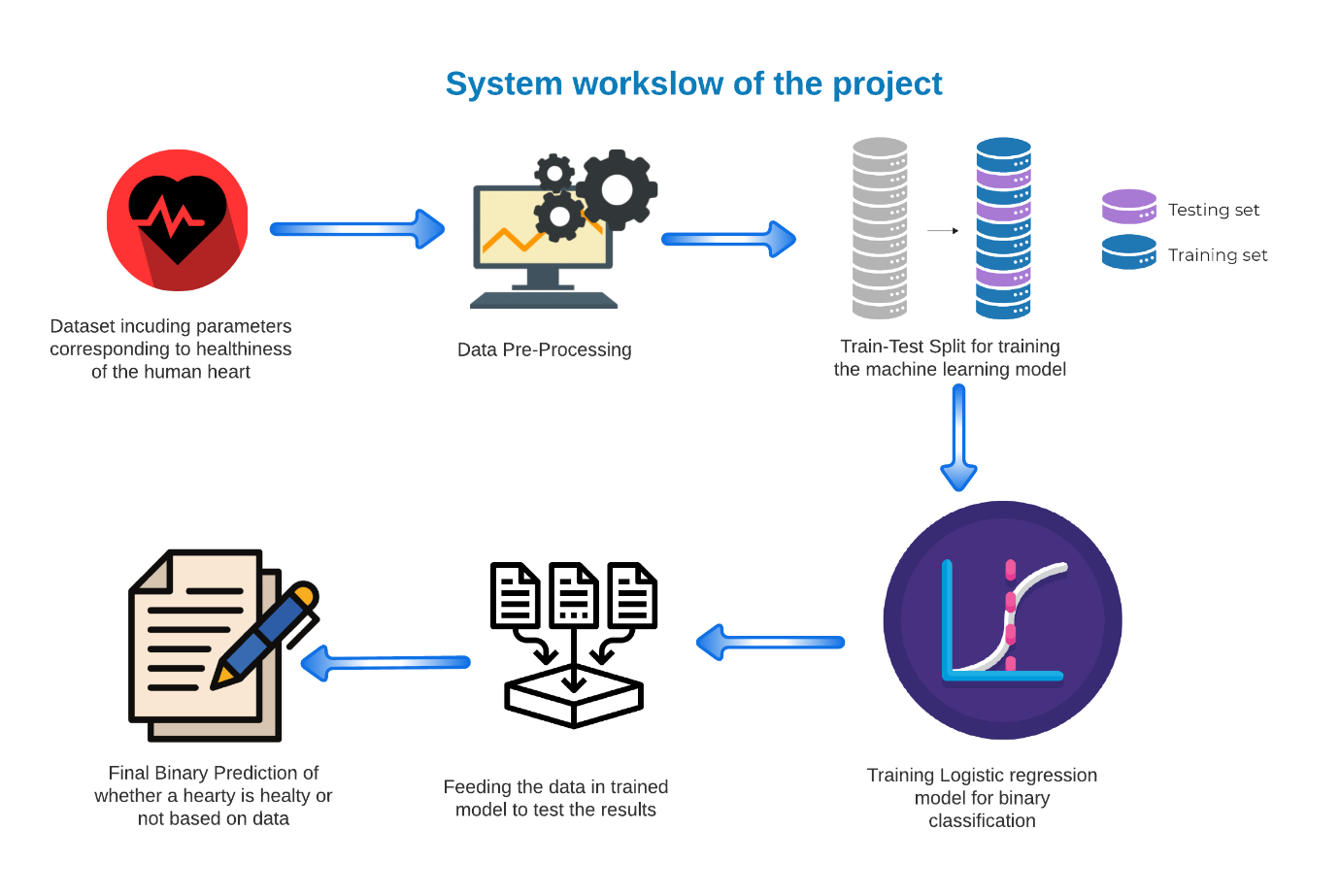
There are 918 data in the dataset. 11 features are included in the dataset.

* Age: age of the patient [years]
* Sex: sex of the patient [M: Male, F: Female]
* ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
* RestingBP: resting blood pressure [mm Hg]
* Cholesterol: serum cholesterol [mm/dl]
* FastingBS: fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
* 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]
* MaxHR: maximum heart rate achieved [Numeric value between 60 and 202]
* ExerciseAngina: exercise-induced angina [Y: Yes, N: No]
* Oldpeak: oldpeak = ST [Numeric value measured in depression]
* ST\_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]

All of these are related to heart conditions that may lead to heart disease. 508 people have heart disease and 410 people do not have heart disease in the dataset. This dataset was created by combining different datasets already available independently but not combined before. In this dataset, 5 heart datasets are combined over 11 common features which makes it the largest heart disease dataset available so far for research purposes.

Dataset link: <https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction>

## System workflow:



## Data Pre-processing:

There are no null values so there is no need to remove any particular rows. Only pre-processing is to encode some columns.

Encoded using Label Encoder to turn string data into int

* Sex: 1 means M(male), 0 means F(female)
* Chest Pain: 2 means NAP, 1 means ATA, 0 means ASY.
* Resting ECG: 2 means ST, 1 means Normal, 0 means LVH
* Exercise Angina: 1 means Y(yes), 0 means N(no)
* ST\_Slope: 2 means up, 1 means flat, 0 means down.

## Methodology:

1. All necessary libraries and dependencies are imported in Google Colab.
2. Dataset is divided into 4:1 ratio for Train: Test validation.
3. X and Y data frame created for feature set and target (binary prediction)
4. Model (logistic regression) is fitted with training and testing set of data.
5. Model evaluation with accuracy score (both training and testing set)
6. Building a predicting system with the trained model. The technique is to reshape the dataset after forming a NumPy array with the dataset. So, if the prediction result is 1, then the result will show the person has heart disease. If the result is 0, then there is no heart disease.

## Extended work

Now there will be a comparison of accuracy based on random state in logistic regression

|  |  |  |
| --- | --- | --- |
| Random State | Accuracy Training Set (%) | Accuracy Testing Set (%) |
| 1 | 85.83106267029973 | 84.23913043478261 |
| 2 | 85.01362397820164 | 85.32608695652173 |
| 3 | 85.69482288828338 | 83.15217391304348 |
| 4 | 86.51226158038146 | 82.6086956521739 |
| 5 | 86.37602179836512 | 84.78260869565217 |
| 6 | 85.14986376021798 | 83.69565217391305 |
| 7 | 85.01362397820164 | 85.32608695652173 |
| 8 | 84.87738419618529 | 86.95652173913044 |
| 9 | 84.74114441416893 | 88.04347826086956 |
| 10 | 85.01362397820164 | 87.5 |

## Result and Decision

As seen from the table, in term of the training set, the best result is given when the random state is at 4, for testing set the best result is for random state 9. And for both the set, average result is best when the random state is 9. In the model training the stratify must be equal to the test set result so that overfitting does not happen. Random state is used to set the seed for the random generator so that we can ensure that the results that we get can be reproduced. Because of the nature of splitting the data in train and test is randomised you would get different data assigned to the train and test data unless you can control for the random factor.