Model Building

Overview

The type of model we need for predicting each type of disease is almost same. We train the model with already available dataset. Hence, we are building the model using supervised learning method. For supervised learning, we have either classification type or regression type.

I was highly biased towards building logistic regression model because the use case we have exactly fits the definition of logistic regression. We need either yes or no (binary classification) for our target dataset i.e., the model should be able to predict either the patient is prone to the disease or not based on one or more independent variables. Hence, I have trained the model using logistic regression method. I have also used decision tree regressor model just for comparison purpose.

Steps for building the model

1. Finding training dataset:

I used the dataset provided in the project specification for building the models.

1. Cleaning Dataset:

Some of the rows of the dataset have null values. Instead of deleting the whole row and decreasing the size of dataset, I filled the null values with mean value of the column instead of 0, to make dataset more accurate for prediction. In some cases, I had to map target variables to either 0 or 1 (binary classification).

1. Splitting dataset:

In order to build a good model, we need to see how the model predicts the accuracy of the model. Hence, I split the dataset into training and testing set using train-test split function from sci-kit learn.

1. Fitting and Predicting:

I fit the train dataset (X\_train and y\_train) to train the model and evaluated how it performs with test dataset.

1. Saving Model:

I saved the model in .sav format using joblib library of python. We can use this model to predict real world data, which we will be getting from Medi-App and Medi-web

Models Use Case

Models are fed into database from where the Medi-App or Medi-Web gains functionality of predicting if the patient is prone to the respective disease or not. It is necessary for the user to input details necessary for predicting where details required for predicting each disease may vary.

Testing Model

Once the model is trained, it is saved in the working directory in different format like .sav format. It should be tested before it is fed into the database. Hence, I created a separate python file called Model\_tests.py. In this file, I loaded the model, prepared data and get the score of the model through which we know the accuracy of the model.