Indians Diabetic Prediction

Predict the onset of diabetes based on diagnostic measures

Data Overview

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. URL for the dataset is https://www.kaggle.com/kumargh/pimaindiansdiabetescsv
The Data Contains Mainly 9 columns and 768 records. Records contain several medical datas and a target varibale to predict whether diabetic present or not.

Objective of work

The objective of the dataset is to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset.

Contents

The datasets consists of several medical predictor variables and one target variable, Outcome. Predictor variables includes the number of pregnancies the patient has had, their BMI, insulin level, age, and so on.

Variables

Variables are divided into two categories: X and y

X contains independet Variables and y contain target variable called dependent variable

Independed variables:

1.Number of times pregnant : Numeric data
2. Glucose Concentration : Numeric
3. Blood Pressure : Numeric
4. Skin thickness : Numeric
5. Insulin : Numeric
6. Body mass index : Numeric
7 .Diabetes Pedigree Function : Numeric

8. Age : Numeric

Dependent Variable

Diabetes outcome : boolean

Approach

1. Import Libraries

Import important libraries like pandas,Numpy,Seaborn,Matplot etc.

2 . Load Data

Load dataset from location path

3. View Data

View data set using Head and Tail and verify whether data is correct or not. Check columns and rows count.

4. Data Preprocessing

Data Cleaning and EDA

1. Check whether have null values or not
2. Check datatypes of each variables
3. Maped boolen to numerical values
4.Check whether data is imbalanced or not.
5. impute Zero values with mean values using Imputer
6. Use heatmap to check whether have null values or not

5. Model Building

Split the dataset to independet and dependent variables and use train_test_split library to seperate data to 70:30 ratio train and test

6. Algorithms

Random Forest Algorithm is used to check the accuracy.

For hyperparameter tuning used RandomizedSearchCv using Xgboost, and also verified with GridsearchCV

Result

There is no any null values in the data set. But have some zero values and had impute these values with mean values using SimpleImputer. Correlation was better. Using Heatmap check the null values present or not. Boolean values maped to numerical values. And finally split the test and train data set and perform training the dataset to predict the test data.

Used Random Forest Algorithm - Accuracy : 77%

For Hyperparameter tuning used RandomizedSearchCV using Xgboost - Accuracy :77%

(when used GridSearchCV for Hyperparameter tuning got only 74% accuracy. So for the best performnace used RandomizedSearchCV).

Checked with other supevised algorithms like SVM,KNN,Logstic Regression but the accuracy was less compared to RF so here i chosen RF algorithm for better performance.