**Readme File**

**Diabetes Prediction Using Machine Learning in Python**

**Machine Learning**

There are 3 main types of machine learning i.e. Supervised Learning, Unsupervised Learning and Reinforcement Learning. As a subset of Artificial Intelligence (AI), machine learning can be used to solve a myriad of problems such as fraud detection, web search results, credit scoring, customer segmentation, email spam filtering, etc.

I decided to grow my machine learning skills by engaging in diabetes prediction. I did this not only for fun and to learn but also to appreciate the essence of machine learning in solving some of the problems that plague humanity. This is, therefore, an interesting project. So, let us do this!

**About the Dataset**

his dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. It is provided courtesy of the Pima Indians Diabetes Database and is available on Kaggle. [Here is the link to the dataset](https://www.kaggle.com/uciml/pima-indians-diabetes-database/data). It consists of several medical predictor variables and one target variable, Outcome. Predictor variables include the number of pregnancies the patient has had, their BMI, insulin level, age, and so on. The dataset has 9 columns as shown below;

* Pregnancies               – Number of times pregnant
* Glucose                     – Plasma glucose concentration a 2 hours in an oral glucose tolerance test
* BloodPressure            – Diastolic blood pressure (mm Hg)
* SkinThickness           – Triceps skinfold thickness (mm)
* Insulin                        – 2-Hour serum insulin (mu U/ml)
* BMI                            – Body mass index (weight in kg/(height in m)^2)
* DiabetesPedigreeFunction – Diabetes pedigree function
* Age                            – Age (years)
* Outcome                    – Class variable (0 or 1) 268 of 768 are 1, the others are 0

**Problem Statement**

This is a classification problem of supervised machine learning. The objective is to predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset.

0 – Absence of Diabetes

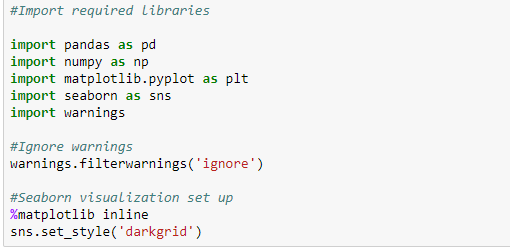
1 – Presence of Diabetes

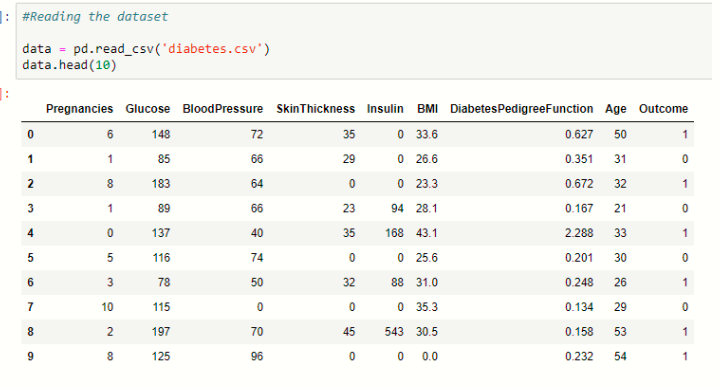
In this project, we will use python3 and Jupyter notebook. Feel free to use your preferred IDE. We will go through the project by importing the dataset, conducting exploratory data analysis to get insights and understanding on how the dataset looks like and then build the model. We will further use Decision Trees, Random Forests, Support Vector Machines and XGBoost.

The 7 steps of machine learning provides the following general framework of steps in supervised machine learning:

* Data Collection
* Data Preparation
* Choosing a model
* Training the model
* Evaluating the model
* Parameter tuning
* Making prediction

So let us begin by importing the required libraries. We will import data analysis libraries (pandas, numpy) and visualization libraries (matplotlib, seaborn). In addition to that, we will also import warnings so that warnings are hidden from the notebook in case there is.





3. check the numbers of rows and columns

Using shape() , its return the value in tuples

4. Let us now check the columns and their data types.

Using info()

from the dataset, 7 columns have int64 data type and 2 columns have float64 data type.

5. Check the null values

Using isnull().sum()

6. # check data is balance or not

Using value\_counts()

7. #Separate input and output from datset

X=data.drop('target',axis=1)# input features

Y=data['target'] # output

# 8. We will now split our dataset before we train it.

X will contain all the Independent variables while y will have the Dependent variable (Outcome).

Using train\_test\_splilt

9. Then apply scaling on output data before train and data

#apply standard scaler for input data training and testing

We have import

from sklearn.preprocessing import StandardScaler

and then last applying machine learning model.

**10. Building the Model :**

We will use models i.e. Logistic Regression ,Random Forests, Decision Trees, Entropy , ADA Boosting ,XGBoost and Support Vector Machine to get the best accuracy score. ‘Accuracy’ metric is used to evaluate models. It is the ratio of the number of correctly predicted instances in a dataset divided by the total number of instances in the dataset. We will proceed further to explore more metrics to determine the best model.

We use all the model and then get the result ,

# We have used 11 different methods the Best result we got is in Decision Tree Entropy max\_depth .74 i.e. 74%.