**DATASETS**

1. Age: This refers to the age of the individual in years. It's a numerical value representing how old a person is.
2. Gender: This indicates the gender of the individual, typically represented as a categorical variable with two values, such as "Male" and "Female".
3. FPG (Fasting Plasma Glucose): FPG is a measure of the concentration of glucose (sugar) in the blood after an overnight fast. It's used to diagnose and monitor diabetes and prediabetes.
4. Chol (Total Cholesterol): Total cholesterol is the total amount of cholesterol in the blood, including both "good" (HDL) and "bad" (LDL) cholesterol.
5. Tri (Triglycerides): Triglycerides are a type of fat found in the blood. High triglyceride levels can increase the risk of heart disease.
6. HDL (High-Density Lipoprotein): HDL is often referred to as "good" cholesterol because it helps remove other forms of cholesterol from the bloodstream, reducing the risk of heart disease.
7. LDL (Low-Density Lipoprotein): LDL is often referred to as "bad" cholesterol because it can build up in the walls of arteries, leading to atherosclerosis and increasing the risk of heart disease.
8. ALT (Alanine Aminotransferase): ALT is an enzyme found mainly in the liver. Elevated levels of ALT in the blood can indicate liver damage or disease.
9. Creatinine: Creatinine is a waste product produced by muscles from the breakdown of a compound called creatine. It is filtered out of the blood by the kidneys and excreted in urine. Measurement of creatinine levels is used to assess kidney function.

These features are commonly used in medical datasets to assess various health indicators and risks. Machine learning models can be trained on such data to predict outcomes like risk of diabetes, heart disease, or kidney disease, among others.

CODE

Diabetes.ipynb

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import accuracy\_score

from sklearn.pipeline import make\_pipeline

from sklearn.preprocessing import StandardScaler

* **These lines import necessary libraries for data manipulation (`pandas`), data visualization (`matplotlib` and `seaborn`), machine learning tasks like splitting data (`train\_test\_split`), evaluating model performance (`accuracy\_score`), and preprocessing data (`StandardScaler`, `make\_pipeline**`).

data = pd.read\_csv("../Datasets/Book1.csv")

* This line reads the dataset from the specified file path using pandas' `read\_csv()` function.

data.info()

* This line prints a concise summary of the dataframe, including the data types and the number of non-null values in each column.

data.describe()

* To generate descriptive statistics of the dataset, it should be `data.describe()`.

# Delete columns directly by specifying their names

columns\_to\_delete = ['sno', 'Name'] # List of column names to delete

data.drop(columns=columns\_to\_delete, inplace=True)

* This code deletes the specified columns (`sno` and `Name`) from the dataset. It uses the `drop()` method of pandas DataFrame to drop columns specified in the `columns\_to\_delete` list. Setting `inplace=True` makes the changes to the DataFrame permanent.

data.isnull().sum()

data.isna().sum()

* These lines count the number of missing values in each column of the DataFrame. `isnull()` and `isna()` are methods to detect missing values, and `sum()` calculates the total number of missing values for each column.

data["Gender"]=data["Gender"].map({"M":1,"F":0})

* This line maps the values in the "Gender" column to numerical values. It assigns 1 to "M" (Male) and 0 to "F" (Female).

data["Diabetes"].value\_counts().plot.bar()

* This line plots a bar chart showing the distribution of values in the "Diabetes" column. It uses `value\_counts()` to count the occurrences of each unique value and then plots the result using `plot.bar()`.

data["Diabetes"].value\_counts()

* This line prints the count of each unique value in the "Diabetes" column.

new\_data = data.fillna(method='ffill', axis=1)

* This line fills missing values in the dataset using forward fill (`ffill`). It propagates the last valid observation forward along the specified axis (in this case, along columns).