**TEAM 16: ARTIFICIAL INTELLIGENCE**

**PROJECT TITLE:** **DIABETES PREDICTION SYSTEM**

**PHASE III: INNOVATION**

**DESCRIPTION:**

A diabetes prediction system is a technology or model that uses data analysis and artificial intelligence (AI) techniques to predict the risk of developing diabetes in individuals or to help manage the condition in those who are already diagnosed. Such systems are valuable for early intervention and personalized care. Here's an overview of how a diabetes prediction system typically works in below

**Sample Program:**

1. **Import libraries:**

**In Python, you use the import keyword to make code in one module available in another. Imports in Python are important for structuring your code effectively.**

Some important libraries known as in python is **Pandas, NumPy**

* **import pandas as pd**
* **from sklearn.model\_selection import train\_test\_split**
* **from sklearn.preprocessing import StandardScaler**
* **from sklearn.linear\_model import LogisticRegression**
* **from sklearn.metrics import accuracy\_score**

1. **Dataset:**

* Dataset is a collection of various types of data stored in a digital format
* Data is the key component of any Machine Learning project. Datasets primarily consist of images, texts, audio, videos, numerical data points, etc., for solving various Artificial Intelligence challenges such as. Image or video classification.
* Dataset contains the attributes, attributes in which it input from the user and gives output

We have to load the dataset into program by in a **csv format**

**Dataset Link:**[**https://www.kaggle.com/datasets/mathchi/diabetes-data-set**](https://www.kaggle.com/datasets/mathchi/diabetes-data-set)

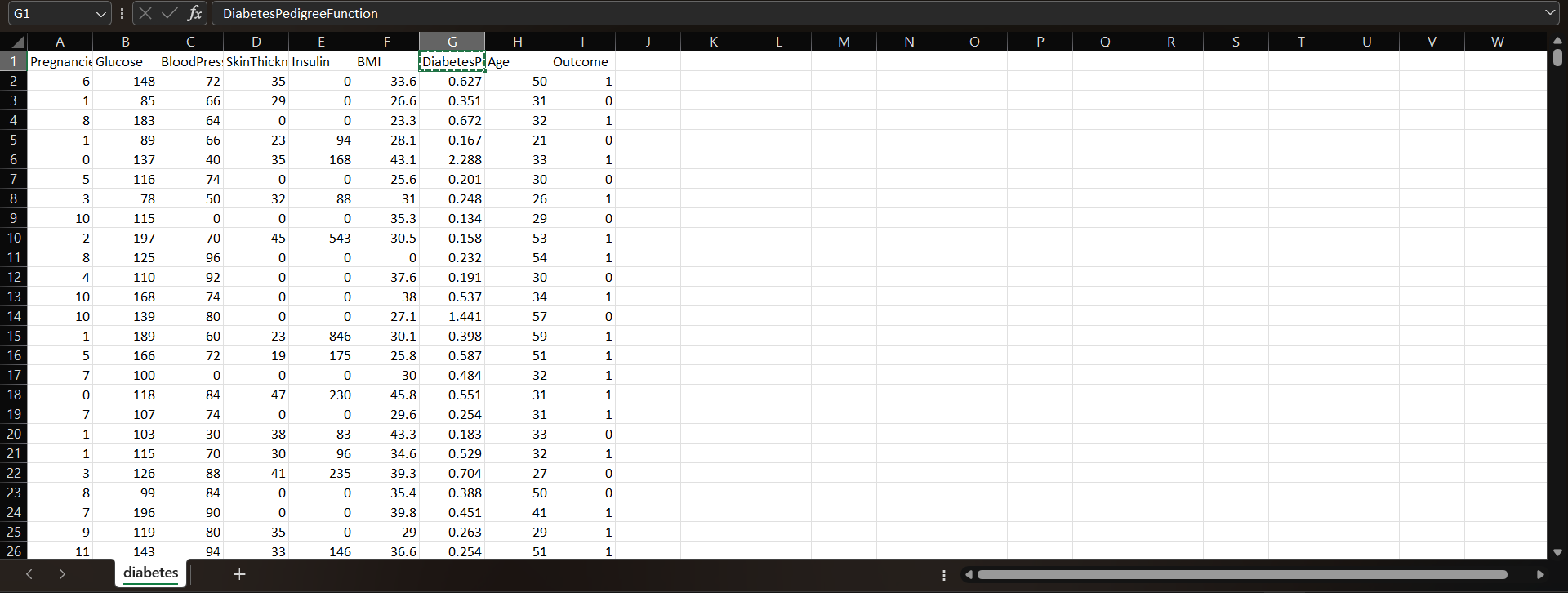
* **Loading the dataset into a program by**

**data = pd.read\_csv("diabetes.csv")**

In our Diabetes prediction system dataset contains attributes such as

* **Pregnancies**
* **Glucose**
* **Blood pressure**
* **Skin Thickness**
* **Insulin**
* **BMI**
* **Diabetes Pedigree Function**
* **Age**
* **Outcome**

**EX:**

****

1. **Data preprocessing:**

**It is used to handle missing values, feature selection, and feature engineering**

**By**

**X = data.drop("diabetes\_label", axis=1)**

**Y= data["diabetes\_label"]**

1. **Split the data into training and testing sets:**

**X\_train, X\_test, y\_train, y\_test = train\_test\_split(features, target, test\_size=0.2, random\_state=42)**

**# Standardize the features**

**scaler = StandardScaler()**

**X\_train = scaler.fit\_transform(X\_train)**

**X\_test = scaler.transform(X\_test)**

**# Create a Logistic Regression model**

**model = LogisticRegression()**

**# Train the model**

**model.fit(X\_train, y\_train)**

**# Make predictions on the test set**

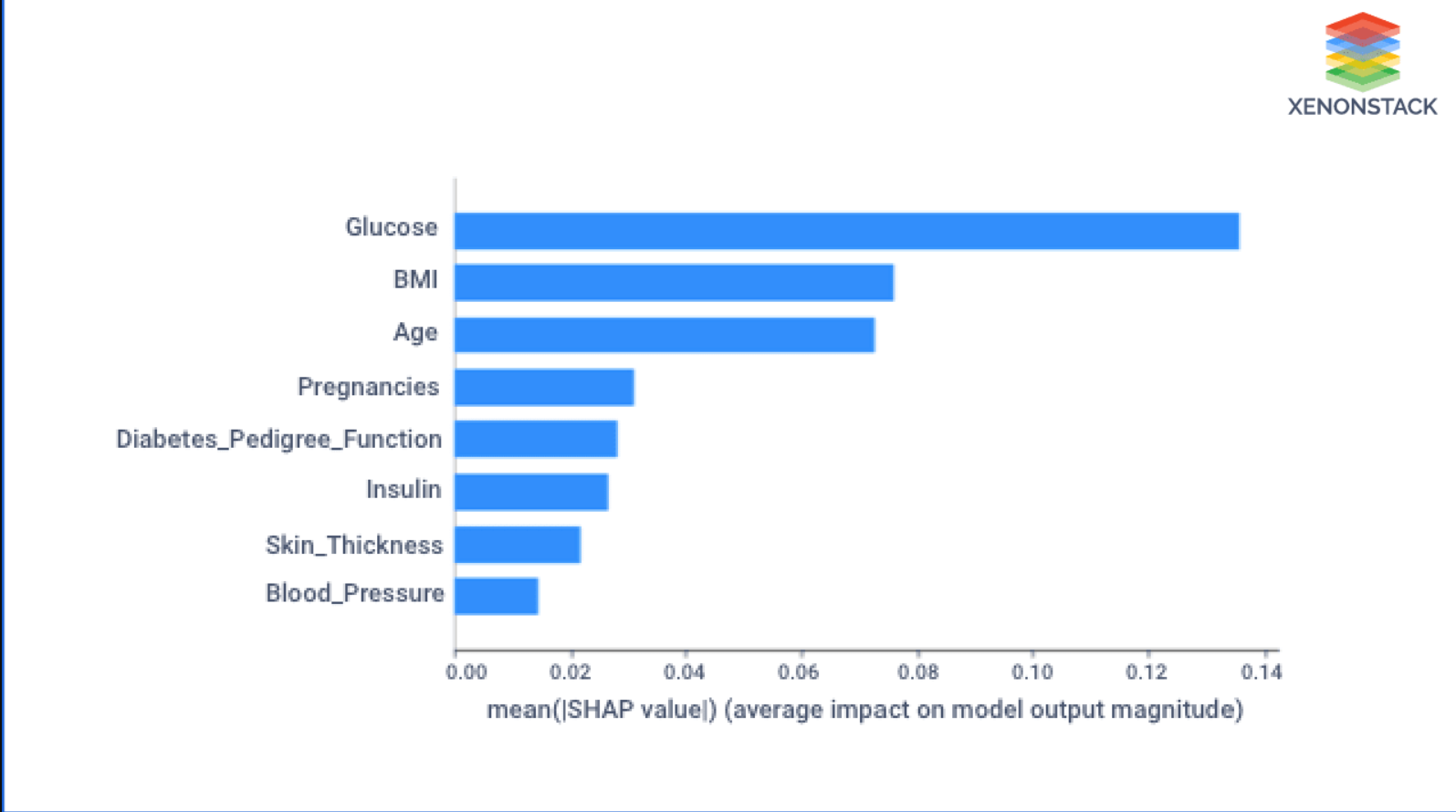
**y\_pred = model.predict(X\_test)**

**# Calculate accuracy**

**accuracy = accuracy\_score(y\_test, y\_pred)**

**print(f"Accuracy: {accuracy:.2f}")**

1. Prediction:



1. Sample Output:

-Age: 35

- BMI: 28.5

- Family History: Yes

- Physical Activity: Moderate

- Glucose Level: 150 mg/dL

- Blood Pressure: 120/80 mmHg

Prediction:

- Likelihood of Diabetes: 0.75 (75%)

- Prediction: Diabetes

Explanation:

Based on the provided patient information, the AI model predicts a 75% likelihood of the patient having diabetes. The prediction is "Diabetes."

Please consult a healthcare professional for further evaluation and diagnosis.