AI Based Diabetes Prediction System

NAME GANESAN.D
REG NO 720421104014

AI_PHASE1 DOCUMENT SUBMISSION

PROJECT : AI Based Diabetes Prediction System

PROBLEM DEFINITION :

The problem is to build an AI-powered diabetes prediction system that uses machine learning algorithms to analyze medical data and predict the likelihood of an individual developing diabetes. The system aims to provide early risk assessment and personalized preventive measures, allowing individuals to take proactive actions to manage their health. Early prediction of diabetes and prediabetes can reduce treatment cost and improve intervention. The development of (pre)diabetes is associated with various health conditions that can be monitored by routine health checkups. This study aimed to develop amachine learning-based model for predicting (pre)diabetes.

DESIGN THINKING :

1. Data Collection: We need a dataset containing medical features such as glucose levels, blood pressure, BMI, etc., along with information about whether the individual has diabetes or not.
2. Data Preprocessing: The medical data needs to be cleaned, normalized, and prepared for training machine learning models.
3. Feature Selection: We will select relevant features that can impact diabetes risk prediction.
4. Model Selection: We can experiment with various machine learning algorithms like Logistic Regression, Random Forest, and Gradient Boosting.

5. Evaluation: We will evaluate the model's performance using metrics like accuracy, precision, recall, F1-score, and ROC-AUC.
6. Iterative Improvement: We will fine-tune the model parameters and explore techniques like feature engineering to enhance prediction accuracy.

PROGRAM :

```
import pandas as pd

import numpy as np

from sklearn.preprocessing import StandardScaler

from sklearn.linear_model  import LogisticRegression

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score, confusion_matrix

import matplotlib.pyplot as plt

import seaborn as sns



import pandas as pd

data = pd.read_csv('/kaggle/input/diabetes-data-set/diabetes.csv')

data.head()



data.describe()



data.isnull().sum()
```

```
data['BMI'] = data['BMI'].replace(0,data['BMI'].mean())

data['BloodPressure'] =
data['BloodPressure'].replace(0,data['BloodPressure'].mean())

data['Glucose'] = data['Glucose'].replace(0,data['Glucose'].mean())

data['Insulin'] = data['Insulin'].replace(0,data['Insulin'].mean())

data['SkinThickness'] =
data['SkinThickness'].replace(0,data['SkinThickness'].mean())



import matplotlib.pyplot as plt

import seaborn as sns

fig, ax = plt.subplots(figsize=(15,10))

sns.boxplot(data=data, width= 0.5,ax=ax,  fliersize=3)



X = data.drop(columns = ['Outcome'])

y = data['Outcome']




from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test =
train_test_split(X,y,test_size=0.25,random_state=0)

X_train.shape, X_test.shape
```

```
import pickle

##standard Scaling- Standardization

def scaler_standard(X_train, X_test):

    #scaling the data

    scaler = StandardScaler()

    X_train_scaled = scaler.fit_transform(X_train)

    X_test_scaled = scaler.transform(X_test)


    #saving the model

    file = open('standardScalar.pkl','wb')

    pickle.dump(scaler,file)

    file.close()


    return X_train_scaled, X_test_scaled


X_train_scaled, X_test_scaled = scaler_standard(X_train, X_test)



X_train_scaled
```

```
log_reg = LogisticRegression()


log_reg.fit(X_train_scaled,y_train)
```