ML-MINOR-MAY

PROBLEM STATEMENT : This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. 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. Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.

PACKAGES:

1.PANDAS Pandas is a Python library that is used for faster data analysis**,** data cleaning, and data pre-processing**.**

2.NUMPY NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.

3.MATPLOTLIB Matplotlib is a Python library for data visualizations. It helps you to mostly plot 2- D dimensional graphs.

4.SKLEARN Scikit-learn is an open source Python library that has powerful tools for data analysis and data mining. It's available under the BSD license and is built on the following machine learning libraries: NumPy, a library for manipulating multi-dimensional arrays and matrices.

EXPLANATION :

As per Machine learning steps :

1.Take the Data

2. Filter the Data

3.Divide input and output

4.Train and test variables

5.Normalise the data

6.Run a classifier

7.Fit the model

8.Predict the output

*To deal with the data we have to import a library called pandas, NumPy .For plotting the data we use matplotlib.

→ For Intake of the Data and forming a data frame we use

```
df = pd.read_csv("/content/diabetes.csv")
```

→We can divide input and output using iloc

Python **iloc()** function enables us to select a particular cell of the dataset, that is, it helps us select a value that belongs to a particular row or column from a set of values of a data frame or dataset.

```
x = df.iloc[:,0:8].values
y = df.iloc[:,-1].values
```

→Here for getting more accurate result we use KNN(Knearestneighbour)

```
from sklearn.neighbors import KNeighborsClassifier
```

→In the view of preparing most accurate model from trail and error method I justified the k value as 3

```
cls = KNeighborsClassifier(n_neighbors=3,metric="euclidean")
```

→Now we should fit the model with the above classifier.

Fit in the sense it is calculating the mean and variance of each of the features present in our data.

```
cls.fit(x,y)
```

→For predicting the accuracy we used our data as an input because we already know the actual output

```
pred_y = cls.predict(x)
```

→Finally we are going to check our machines accuracy.

```
from sklearn.metrics import accuracy_score
accuracy_score(pred_y,y)
```

→we got nearly 85.9% accurate model.

NOTE: I have also tried this using logical regression, but accuracy of the model is nearly 65 % so I choose the method of KNN to solve this problem.

CONCLUSION:

Using KNN we can prepare more accurate model as above.

We have prepared our model which can test the diabetes with 85.9 % of accuracy.

YERRAM DEEKSHITHKUMAR