9/5/24 week-6

Logistic Regression

```
import numpy as np
import Pandas as pd
import matplotlib.pyplot as plt
import Seaborn as sns
from sklearn.dataset import load_diabetis
from sklearn model_selection import train_test_split
from sklearn.linear_model import Logistic Regression
from sklearn.matrix import accuracy_score,
        classification_report, confusion_matrix,

diabetis = load_diabetis()
x, y = diabetis.data, diabetis.target
y_binary = (y > np.median(y)).astype(int)
x_train, x_test, y_train, y_test = train_test_split
    (x, y_binary, test_size = 0.2, random_state = 42)

scaler = standard scaler()
x_train = scaler.fit_transform(x_train)
x_test = scaler.transform(x_test)

model = Logistic Regression()
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy : {:.2f}%".format(accuracy*100)
print("confusion matrix: ", confusion_matrix(y_test, y_pr
print("classification method", classification_report(y_test, y_pred
```

Confusion matrix

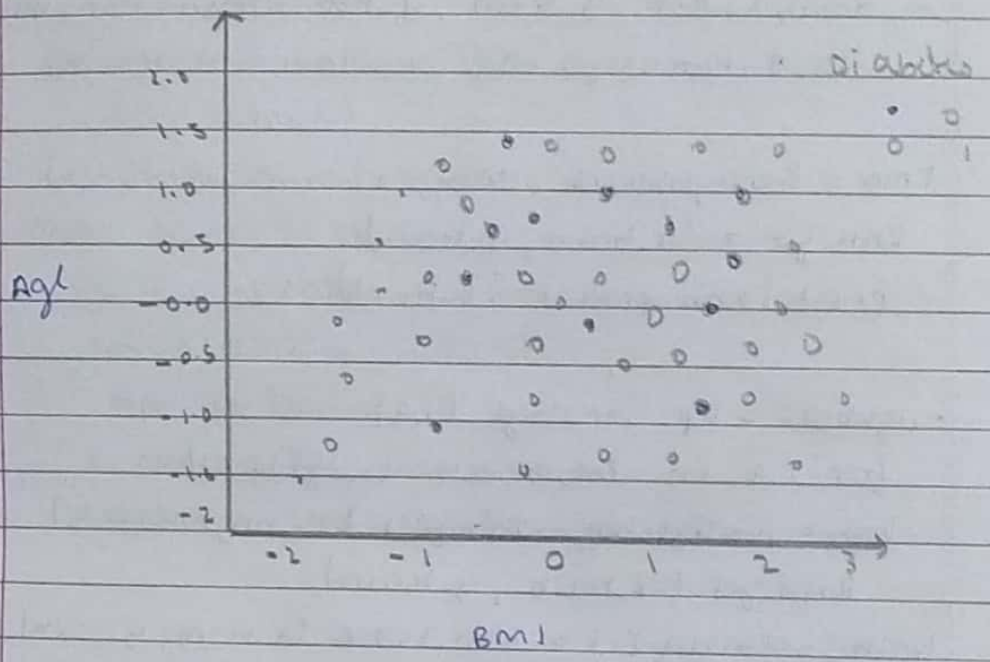
[[36 13]

[11 29]]

Sns. scatter plot (x=x_test[:,2], y=x_test[:,8], hue palette={0:'blue',1:'red'}, marker='o')

plt.legend(title='Diabetes', loc='upper right')

plt.show



* KNN

```
from sklearn neighbours import KNeighbour classifier
from sklearn model_selection import train_test_split
from sklearn datasets import load_iris

iris Data = load_iris()
    x = iris data . data
    y = iris Data. target
x_train, x_test , y_train , y_test = train_test_split
    (x, y , test_size =0.2 , random_state = 42)

KNN = KNeighbours classifier ( n_neighbors =7)
 knn. fit = (x_train , y_train)
 print (knn .predict (x_test))

neighbors = np. arrange (1,9)
  for i,k in enumerate (neighbors)
  knn = KNeighbors classifier (n_ neighbors=k)
    knn. fit ( x_train , y_train)
train_accuracy [i] = knn. score (x_train, y_train)
test_accuracy [i] = knn . score (x_test , y_test)

Plt. legend ()
Plt. xlabel ('n_neighbors')
Plt. ylabel ('Accuracy')
 Plt. show()
```