**Page.No.:**

**A PYTHON PROGRAM TO IMPLEMENT KNN MODEL**

**Ex.No.:**

**Date of Experiment:**

**Date of Submission:**

**AIM:-**

To implement a python program using a KNN Algorithm in an model.

**ALGORITHM:-**

Step1: Import all the other necessary libraries(numpy as np, matplotlib.pyplot as plt and sklearn.tree,pandas as pd and seaborn as sns).

Step2: Select the number K of the neighbors.

Step3: Calculate the Euclidean distance of **K number of neighbors of data points.**

Step4: Take the K nearest neighbors as per the calculated Euclidean distance.

Step5: Among these k neighbors, count the number of the data points in each category.

Step6: Assign the new data points to that category for which the number of the neighbor is maximum.

Step7: Plot the graph “X” and “y” the values tested and predicted using seaborn.scatterplot() function.

Step8: Print the confusion matrix of the model to know the accuracy of the model with support values for each class.

**IMPLEMENTATION:-**

import numpy as np

import pandas as pd

from matplotlib import pyplot as plt

from sklearn.datasets import load\_breast\_cancer

**Page.No.:**

from sklearn.metrics import confusion\_matrix

from sklearn.neighbors import KNeighborsClassifier

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

import seaborn as sns

sns.set()

breast\_cancer = load\_breast\_cancer()

X = pd.DataFrame(breast\_cancer.data, columns=breast\_cancer.feature\_names)

X = X[['mean area', 'mean compactness']]

y = pd.Categorical.from\_codes(breast\_cancer.target, breast\_cancer.target\_names)

y = pd.get\_dummies(y, drop\_first=True)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, random\_state=1)

import warnings

warnings.filterwarnings('ignore')

knn = KNeighborsClassifier(n\_neighbors=5, metric='euclidean')

knn.fit(X\_train, y\_train)

KNeighborsClassifier(metric='euclidean')

y\_pred = knn.predict(X\_test)

sns.scatterplot(

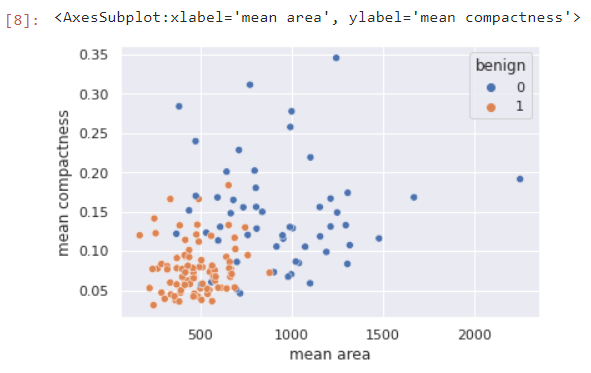
x='mean area',

y='mean compactness',

hue='benign',

data=X\_test.join(y\_test, how='outer')

)



plt.scatter(

X\_test['mean area'],

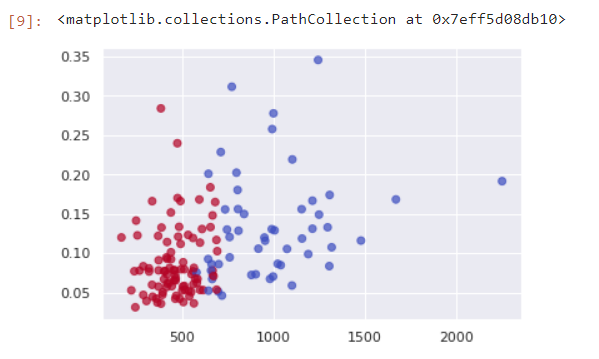
X\_test['mean compactness'],

c=y\_pred,

cmap='coolwarm',

alpha=0.7

)



confusion\_matrix(y\_test, y\_pred)

array([[42, 13],

[ 9, 79]])

**RESULT:-**

Thus the python program to implement KNN model has been successfully implemented and the results have been verified and analyzed.