RESULT : Thus the above execution of the algorithm has been successfully executed.

EX.NO :10

:

IMPLEMENTING ARTIFICIAL NEURAL NETWORKS FOR AN APPLICATION USING PYTHON - CLASSIFICATION

AIM :

To implementing artificial neural networks for an application in classification using python.

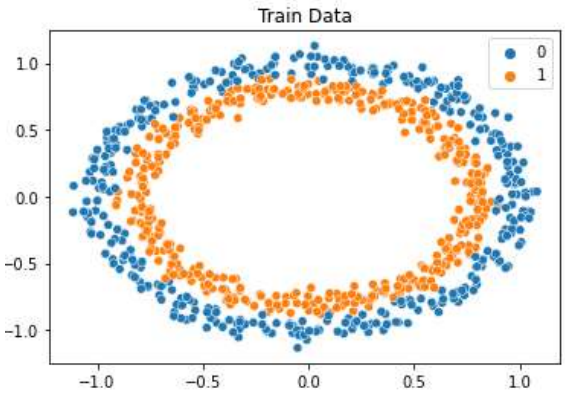
Source Code :

```
from sklearn.neural_network import MLPClassifier
from sklearn.model_selection import train_test_split
from sklearn.datasets import make_circles
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

X, y = make_circles(n_samples=1000, noise=0.05)

ns.scatterplot(X_train[:,0], X_train[:,1], hue=y_train)
```

```
plt.title("Train Data")
plt.show()
```



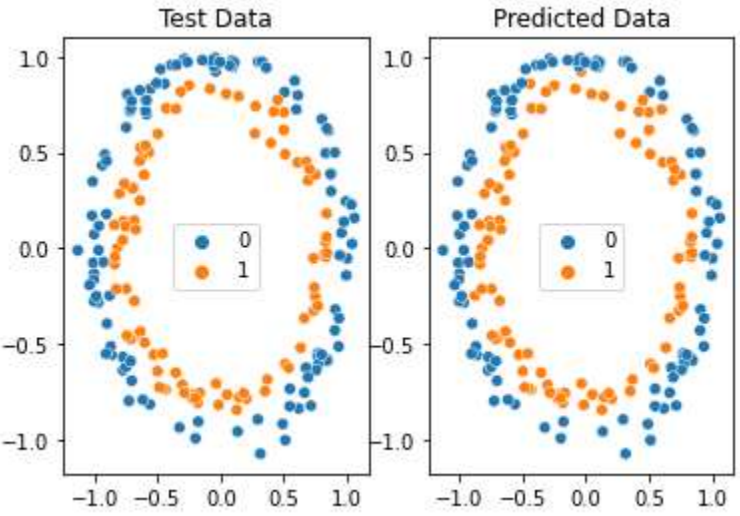

```
clf = MLPClassifier(max_iter=1000)
clf.fit(X_train, y_train)
print(f"R2 Score for Training Data = {clf.score(X_train, y_train)}")

print(f"R2 Score for Test Data = {clf.score(X_test, y_test)}")

y_pred = clf.predict(X_test)

fig, ax =plt.subplots(1,2)
sns.scatterplot(X_test[:,0], X_test[:,1], hue=y_pred, ax=ax[0])
ax[1].title.set_text("Predicted Data")
sns.scatterplot(X_test[:,0], X_test[:,1], hue=y_test, ax=ax[1])
ax[0].title.set_text("Test Data")
plt.show()
```

OUTPUT : Thus the above execution of the algorithm has been successfully executed.



EX.NO :11

:

IMPLEMENTING ARTIFICIAL NEURAL NETWORKS FOR AN APPLICATION USING PYTHON - REGRESSION

AIM :

To implementing artificial neural networks for an application in Regression using python.

SOURCE CODE :

```
from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import train_test_split
from sklearn.datasets import make_regression
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline


X, y = make_regression(n_samples=1000, noise=0.05, n_features=100)

X.shape, y.shape // ((1000, 100), (1000,))
```