Experiment No.: 10 Implementing Artificial Neural Networks for an application using Python Classification

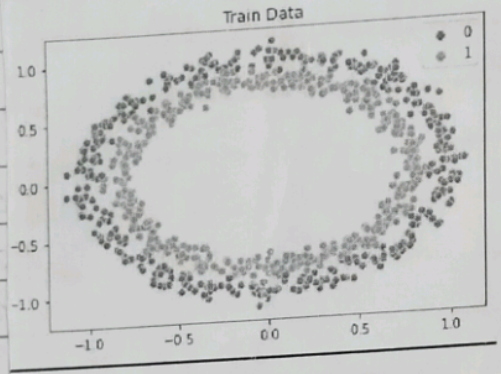
Aim:

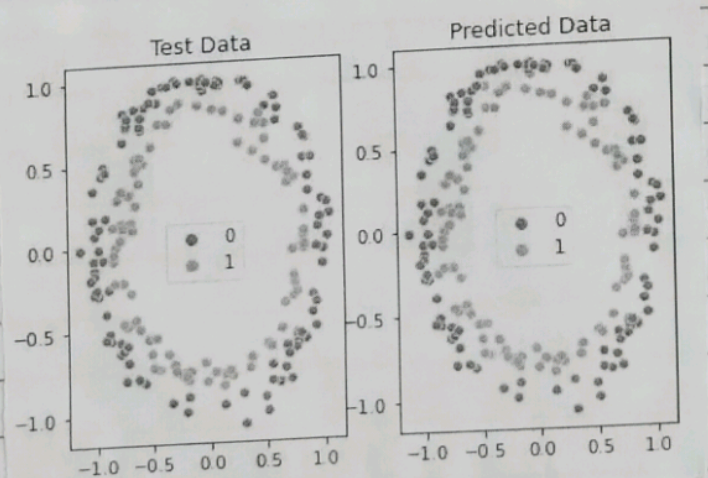
To implementing artificial neural networks for an application in classification using python

Source code:

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

X_train, y_train = make_circles (n_sample=700, noise=0.05)
X_test, y_test = make_circles (n_samples=300, noise=0.05)
sns.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_test,
                                        y_test)}')

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[0].title.set_text("Test Data")
plt.show()
```





Result:

Thus, the program was successfully executed and the output is verified.