IMPLEMENTATION

EX : 8
Date :

ARTIFICIAL NEURAL NETWORKS FOR AN APPLICATION USING PYTHON - REGRESSION

Aim:

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

code:

```
from sklearn.neural_network import
                                MLPRegressor
from sklearn.model_selection import train_
                                test_split
from sklearn.datasets import make_regressor
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,))
X_train, X_test, y_train, y_test =
train_test_split(X,y, test_size=0.2, shuffle=
True, random_state=42)
clf = MLPRegressor(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)}").
```

Output:

R2 score for Test data = 0.9686558466621524

Result:

The program for implementation artificial Neural Networks for an application is successfully implemented.

EX: 9.

Date:

Aim:

To classify dataset using

code:

```
from google.
drive mou
import pa
import nu
import ma
dataset = pd
mydrive[s
X = dataset
y = datas
from skl
train_test
X_train, X
test_split

from sklea
sc = stand
X_train =
X_test =

from skle
cm = confu
print(cm)
```