

Exp. No: 10
Date _____

Implementing Artificial Neural networks for an Application using python - Regression,

Aim:

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

Algorithm:

1. Prepare and split Data: Load or generate a database, then split it into training and test-cases
2. Initialize the model and setup ANN using MLP Regressor.
3. Train the Model and fit to training data, allowing it to learn from pattern in data.
4. Use R^2 Score and other metrics to assess the performance on the test set.

Program:

```
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
```

Date _____
Page _____

$X, Y = \text{make_regression}(n_samples = 1000,$
 $\text{noise} = 0.05, n_features = 100)$

$X.\text{shape}, Y.\text{shape} = (1000, 100), (1000, 1)$

$X_train, X_test, Y_train, Y_test =$
 $\text{train_test_split}(\text{X}, \text{Y}, \text{test_size} = 0.2, \text{shuffle} =$

$\text{True}, \text{random_state} =$

$\text{clf} = \text{MLPRegressor}(\text{max_iter} = 1000)$

$\text{clf.fit}(X_train, Y_train)$

$\text{Print}(\text{"R}_2 \text{ Score for Training Data: "}$
 $\text{clf.score}(X_train, Y_train)\text{"})$

$\text{Print}(\text{"R}_2 \text{ score for Test Data: "}$
 $\text{clf.score}(X_test, Y_test)\text{"})$

Output:

$R_2 \text{ Score for Test Data} =$

0.9686558466621527

Result:

Thus we implement artificial intelligence
neural networks for an application
in regression using python.