File Name:mgLab3test2.ipynb

Name:Manavi Uttam Ghorpade

Course:611

Date:2/27/2022

Program Description:program for TEST 2 – kNN REGRESSOR

```
import numpy as np
from sklearn import datasets
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
from sklearn import metrics
from sklearn.neighbors import KNeighborsRegressor
from mglearn import datasets
```

```
pip install mglearn
```

```
Requirement already satisfied: mglearn in /usr/local/lib/python3.7/dist-packages (0.
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: cycler in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: pillow in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: imageio in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: joblib in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/loca
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: scipy>=1.1.0 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist
```

```
run Knn.py
```

```
<Figure size 432x288 with 0 Axes>
```

```
run preamble.py
```

Testing sklearn's regressor with wave dataset

```
X, y = mglearn.datasets.make_wave(n_samples=40)
# split the wave dataset into a training and a test set
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)
# instantiate the model and set the number of neighbors to consider to 3
reg = KNeighborsRegressor(n_neighbors=3)
# fit the model using the training data and training targets
```

```
# fit the model using the training data and training targets
reg.fit(X_train, y_train)
reg.predict(X_test)
print("Score using sklearn knn algorithm: {:.2f}".format(reg.score(X_test, y_test)))
```

```
Score using sklearn knn algorithm: 0.83
```

Testing our Custom Knn with wave dataset

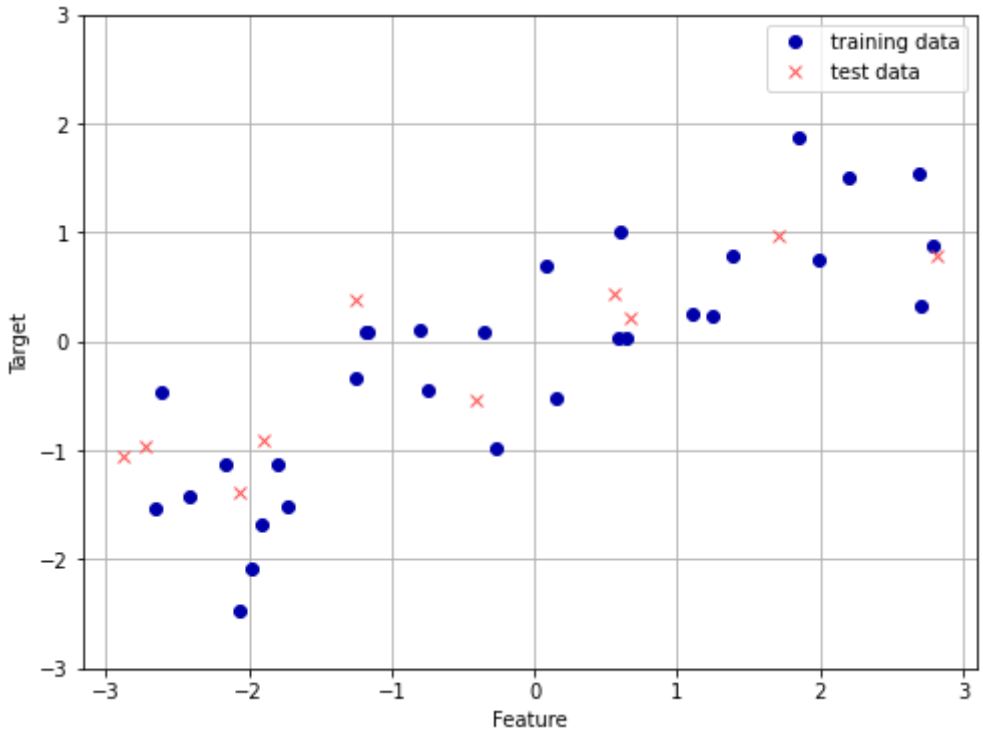
```
# instantiate the model and set the number of neighbors to consider to 3
classifier=knnBuild("regression",3)
# fit the model using the training data and training targets
classifier.knnFit(X_train, y_train)
result=classifier.knnPredict(X_test)
score = metrics.r2_score(y_test, result)
print("Accuracy using the built in knn: {0:f}".format(score))
```

```
Accuracy using the built in knn: 0.834417
```

Plotting Dataset

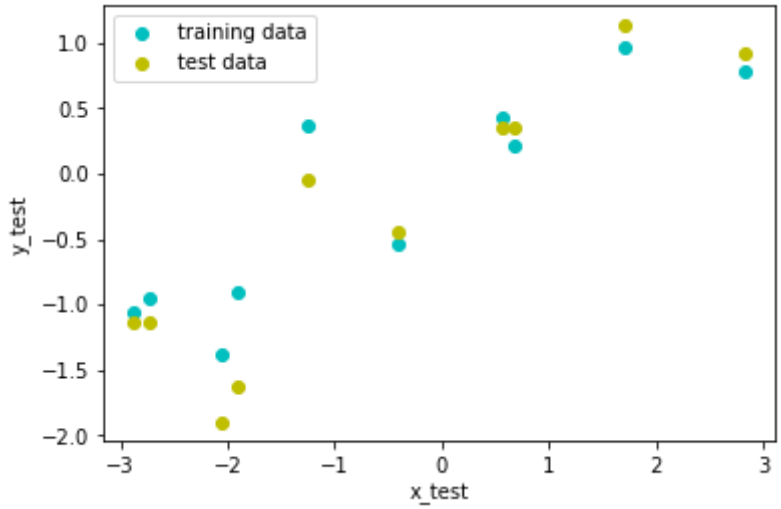
visualization of Data

```
plt.figure(figsize=(8,6))
plt.plot(X_train,y_train, 'o')                    # plot the training data
plt.plot(X_test,y_test, 'x')                      # plot the test data
plt.ylim(-3, 3)
plt.xlabel("Feature")
plt.ylabel("Target")
plt.legend(['training data', 'test data', 'model'])
plt.grid(True)
```



```
#plotting training data and test data with our output
plt.scatter(X_test,y_test,c='c')
plt.scatter(X_test,result,c='y')
plt.ylabel('y_test')
plt.xlabel('x_test')
plt.legend(['training data', 'test data', 'model'])
```

```
<matplotlib.legend.Legend at 0x7f99940eb810>
```



✓ 0s completed at 11:34 PM