

SIEĆ FEEDFORWARD MLP

WIELOWARARSTWOWA SIEĆ NEURONOWA ZADANIA

Korzystając z biblioteki **sklearn** (<http://scikit-learn.org>), wykonaj następujące zadania:

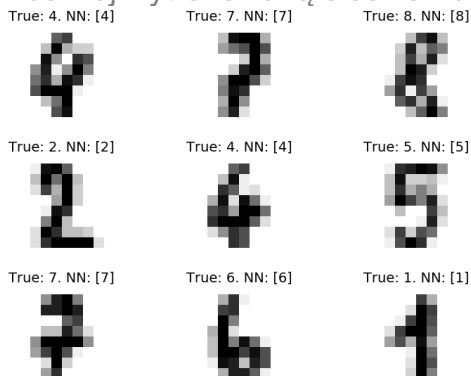
1. Załaduj dane „digits”

```
from sklearn import datasets
digits = datasets.load_digits()
```
2. Zwizualizuj przykładowy element ze zbioru przykładów

```
import matplotlib.pyplot as plt
plt.figure(1)
plt.imshow(digits.images[0], cmap=plt.cm.gray_r)
plt.show() #pamiętaj o zakomentowaniu wykonując dalsza część lab
```
3. Podziel dane na zbiór uczący i testowy

```
from sklearn.model_selection import train_test_split
digits_learn, digits_test, target_learn, target_test =
train_test_split(digits.data, digits.target, test_size=0.005)
```
4. Wytrenuj sieć MLP algorytmem wstecznej propagacji błędów z logistyczną funkcją aktywacji na danych uczących korzystając z klasy **MLPClassifier**

```
from sklearn.neural_network import MLPClassifier
neural_network = MLPClassifier(activation='logistic',
hidden_layer_sizes=(50,30,10))
neural_network.fit(digits_learn, target_learn)
```
5. Zwizualizuj wytrenowaną sieć na zbiorze testowym



```
plt.figure(1)
for i in range(0, len(target_test)):
    plt.subplot('33'+str(i+1))
    plt.imshow(digits_test[i,:].reshape(8,8), cmap=plt.cm.gray_r)
    plt.title('True: ' + str(target_test[i]) + '. NN: '
'+str(neural_network.predict(digits_test[i,:])), fontsize=7)
    plt.axis('off')

plt.tight_layout()
plt.show() #pamiętaj o zakomentowaniu wykonując dalszą część lab
```

6. Oceń jakość wytrenowanego klasyfikatora korzystając z macierzy konfuzji (ang. Confusion matrix) oraz względnej liczby poprawnie rozpoznanych liczb.

```
from sklearn.metrics import confusion_matrix
print(confusion_matrix(target_test,
neural_network.predict(digits_test)))
```

```
from sklearn.metrics import accuracy_score
print(accuracy_score(target_test,
neural_network.predict(digits_test)))
```

DODATKOWE:

Przetestuj działanie sieci MLP na bazie „MNIST” (ręcznie pisane cyfry 0-9, 28x28 pixels, ~70000 przykładów) wyznaczając macierz konfuzji. Zwizualizuj wybrane elementy ze zbioru testowego oraz przedstaw przykładowe rozpoznanie znaków przez sieć neuronową.

```
from sklearn.datasets import fetch_mldata
mnist = fetch_mldata("MNIST original")
```

Przykładowe cyfry z bazy MNIST:



W celu zwizualizowania obrazków ze zbioru przykładów można wykonać:

```
import matplotlib.pyplot as plt
plt.figure(2)
plt.imshow(mnist.data[0].reshape(28,28), cmap=plt.cm.gray_r)
plt.show() #pamiętaj o zakomentowaniu wykonując dalszą część lab
```

Podpowiedź: Zacznij od sieci o małej liczbie warstw ukrytych (np. (50, 10))