

assign5_2_digit_classification

May 15, 2019

```
In [100]: %matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from scipy.io import loadmat
from joblib import Parallel, delayed
import pandas as pd
```

```
In [3]: data = loadmat('digit.mat')
train = data['X']
test = data['T']
```

```
In [4]: print("Train data: {}".format(train.shape))
print("Test data: {}".format(test.shape))
```

```
Train data: (256, 500, 10)
Test data:  (256, 200, 10)
```

```
In [64]: n_class = train.shape[2]
```

```
reshape_x = lambda data: data.reshape(data.shape[0], data.shape[1]*(data.shape[2]))
```

```
def reshape_train_y(data, k):
    zeros = np.zeros(n_class, dtype=np.float32)
    zeros[k] = 1.
    data_y = np.tile(zeros, data.shape[1])
    return data_y
```

```
def reshape_test_y(data):
    arange = np.arange(n_class)
    data_y = np.tile(arange, data.shape[1])
    return data_y
```

```
In [61]: train_x = reshape_x(train)
test_x = reshape_x(test)
test_y = reshape_test_y(test)
```

```
In [27]: def calc_design_matrix(x, c, h=0.3):
    return np.exp(-np.linalg.norm(x[:, :, None] - c[:, None, :], axis=0) / (2 * h ** 2))
```

```
In [67]: def ls_classify(i):
    train_y = reshape_train_y(train, i)
    k_train = calc_design_matrix(train_x, train_x)
    theta = np.reshape(np.linalg.solve(k_train.T.dot(k_train), k_train.T.dot(train_y[:, None])), -1)
```

```

k_test = calc_design_matrix(test_x, train_x)
pred_y = k_test.dot(theta)
return pred_y

```

```

In [68]: pred_y_list = Parallel(n_jobs=n_class)([delayed(ls_classify)(i) for i in range(n_class)])
pred_Y = np.array(pred_y_list)
pred_y = np.argmax(pred_Y, axis=0)

```

```

In [98]: acc = float(np.sum(pred_y == test_y)) / len(test_y)
print acc

```

0.9645

```

In [109]: pred_matrix = {i: {j: 0 for j in range(n_class)} for i in range(n_class)}
for i, j in zip(pred_y, test_y):
    pred_matrix[i][j] += 1

```

```

In [113]: pd.DataFrame(pred_matrix)

```

```

Out[113]:
   0  1  2  3  4  5  6  7  8  9
0 200  0  0  0  0  0  0  0  0  0
1  0 193  1  0  0  0  2  3  1  0
2  0  0 192  0  3  0  2  2  1  0
3  1  0  0 189  1  3  0  0  6  0
4  0  3  4  0 187  0  1  1  2  2
5  0  2  0  0  2 195  0  0  0  1
6  0  0  0  3  0  0 192  2  3  0
7  0  1  4  0  3  0  0 186  3  3
8  0  0  0  2  0  0  1  0 197  0
9  0  1  1  0  0  0  0  0  0 198

```