18CS10069_Q10

October 23, 2021

[1]:
```
# import and loading data
import numpy as np
from matplotlib import pyplot
from keras.datasets import mnist
from collections import defaultdict, Counter
from sklearn.cluster import KMeans
from sklearn.metrics import accuracy_score, confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

(trainX, trainy), (testX, testy) = mnist.load_data()
print('Train: X=%s, y=%s' % (trainX.shape, trainy.shape))
print('Test: X=%s, y=%s' % (testX.shape, testy.shape))
```

```
Train: X=(60000, 28, 28), y=(60000,)
Test: X=(10000, 28, 28), y=(10000,)
```

[2]:
```
# create the training dataset
def create_dataset(train_labels, num=1000):
    image_ids = defaultdict(lambda : 0)
    final_images, final_labels = [], []

    for i in range(len(train_labels)):
        label = train_labels[i]
        if image_ids[label] < num:
            final_images.append(i)
            final_labels.append(label)
            image_ids[label] += 1

        if sum(list(image_ids.values())) == 10 * num:
            assert final_labels == [train_labels[x] for x in final_images]
            return final_images, final_labels

# vectorization of images
def vectorization(final_images, trainX):
    vector_length = trainX[0].shape[0] * trainX[0].shape[1]
    final_trainX = np.zeros((len(final_images), vector_length))
```

```
    for i in range(len(final_images)):
        id = final_images[i]
        final_trainX[i] = trainX[i].reshape(1, -1)/255.0
    return final_trainX
```

[3]:
```
# train set creation
final_images, final_labels = create_dataset(trainy)
final_trainX = vectorization(final_images, trainX)

# test set creation
test_images, test_labels = create_dataset(testy, num=100)
final_testX = vectorization(test_images, testX)
```

[4]:
```
# adding the bias term set
final_trainX = np.array(final_trainX)
final_labels = np.array(final_labels)
bias = np.ones((final_trainX.shape[0], 1))
trainX = np.hstack((final_trainX, bias))

# adding the bias term to test set
final_testX = np.array(final_testX)
test_labels = np.array(test_labels)
bias = np.ones((final_testX.shape[0], 1))
testX = np.hstack((final_testX, bias))

# print the sizes
print(f'Size of train set: {trainX.shape}')
print(f'Size of test set: {testX.shape}')
```

```
Size of train set: (10000, 785)
Size of test set: (1000, 785)
```

[5]:
```
# obtain parameter for a given digit (class)
def get_parameters(X, Y, digit):
    Y_target = [1 if y == digit else -1 for y in Y]
    theta = np.linalg.pinv(X).dot(Y_target)
    return theta
```

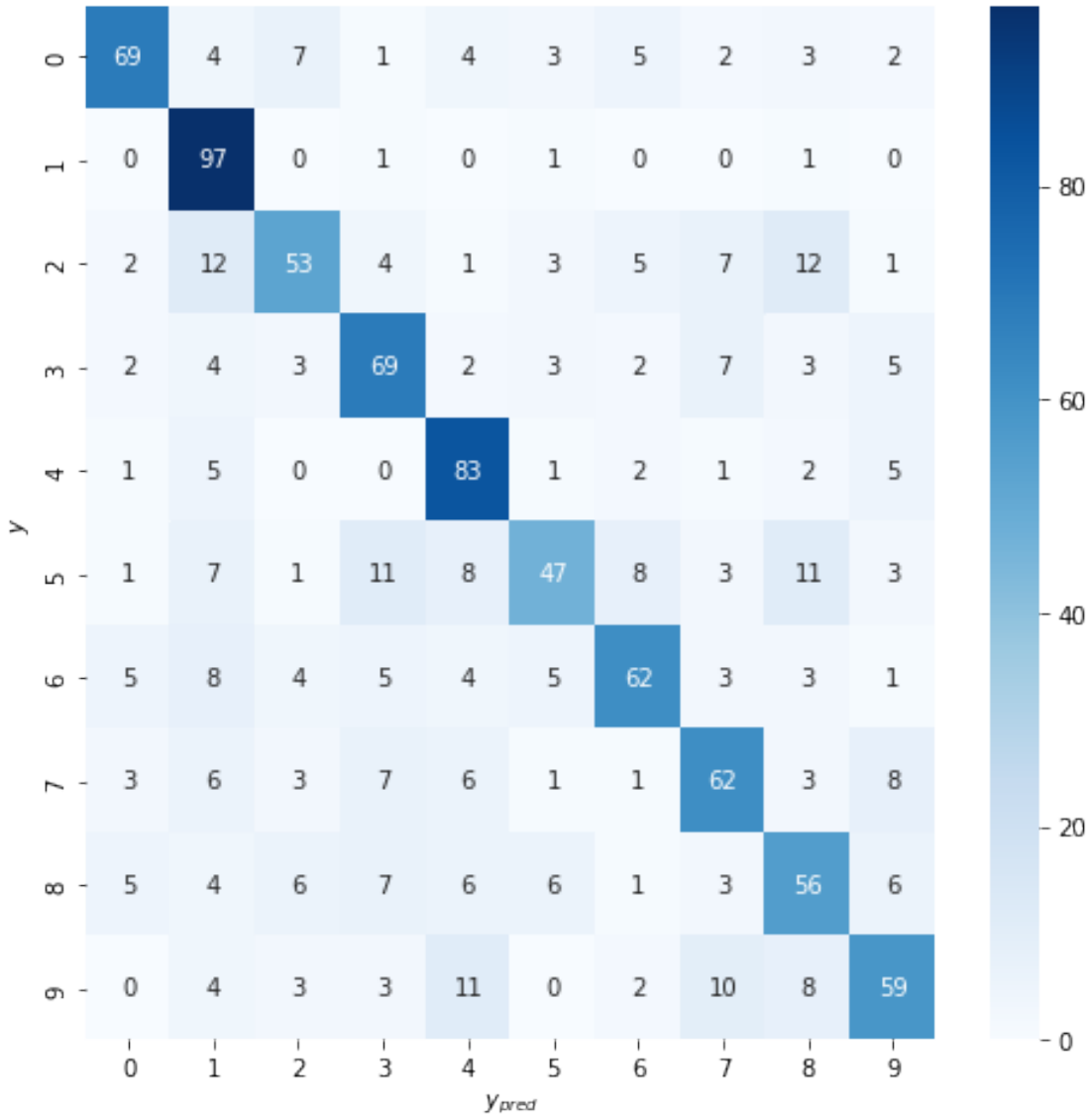
[6]:
```
# collect all the parameters for each digit
parameters = []
for digit in range(10):
    theta = get_parameters(trainX, final_labels, digit)
    parameters.append(theta)
parameters = np.array(parameters)
```

[7]:
```
# predict function
def predict(x):
    y = []
    for digit in range(10):
```

```
        f = np.dot(x.transpose(), parameters[digit])
        y.append(f)
    return y.index(max(y))

y_pred = np.array([predict(x) for x in testX])
```

[8]:
```
# confusion matrix
cm = confusion_matrix(test_labels, y_pred, labels=range(10))
plt.figure(figsize=(8,8));
sns.heatmap(cm, annot=True, fmt='g', cmap="Blues");
plt.xlabel("$y_{pred}$");
plt.ylabel("$y$");
```



[9]:
```
acc = sum([cm[digit][digit] for digit in range(10)])/1000
print(f"Accuracy: {acc * 100}%")
```

```
Accuracy: 65.7%
```

[10]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

[14]:
```
!cd /content/drive/MyDrive/LAAIML2021
# !sudo apt-get install texlive-xetex texlive-fonts-recommended␣
 ↪texlive-generic-recommended
!jupyter nbconvert --to pdf '/content/drive/MyDrive/LAAIML2021/18CS10069_Q9.
 ↪ipynb'
```

```
[NbConvertApp] Converting notebook
/content/drive/MyDrive/LAAIML2021/18CS10069_Q9.ipynb to pdf
[NbConvertApp] Support files will be in 18CS10069_Q9_files/
[NbConvertApp] Making directory ./18CS10069_Q9_files
[NbConvertApp] Making directory ./18CS10069_Q9_files
[NbConvertApp] Making directory ./18CS10069_Q9_files
[NbConvertApp] Making directory ./18CS10069_Q9_files
[NbConvertApp] Writing 37203 bytes to ./notebook.tex
[NbConvertApp] Building PDF
[NbConvertApp] Running xelatex 3 times: [u'xelatex', u'./notebook.tex',
'-quiet']
[NbConvertApp] Running bibtex 1 time: [u'bibtex', u'./notebook']
[NbConvertApp] WARNING | bibtex had problems, most likely because there were no
citations
[NbConvertApp] PDF successfully created
[NbConvertApp] Writing 66552 bytes to
/content/drive/MyDrive/LAAIML2021/18CS10069_Q9.pdf
```