DIGITS' IMAGE CLASSIFICATION

Github link to the jupyter notebook:
https://github.com/ashutosh-999/dig_img_cls_jn.git

SCREENSHOTS

1. **Libraries used:**

```
import os
import torch
from torch import nn

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import random
import torchvision
from torchvision import datasets
from torchvision.transforms import ToTensor
```

2. **Optimizer and Loss Function used**:

```
loss_fn = nn.CrossEntropyLoss()
optimizer = torch.optim.SGD(params = model.parameters(),
                            lr = 0.1)
acc_fn = Accuracy(task = "multiclass", num_classes = 10).to(device)
```

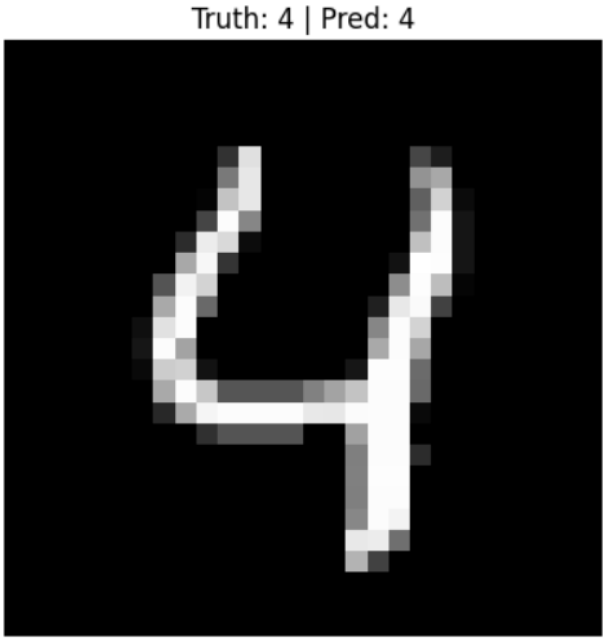
3. **Dataset used:**

```
train_data = datasets.MNIST(root = "data",
                            train = True,
                            download = True,
                            transform = ToTensor(),
                            target_transform = None
                            )

test_data = datasets.MNIST(root = "data",
                           train = False,
                           download = True,
                           transform = ToTensor(),
                           )
```

4. **Accuracy of the model:**

```
100% 5/5 [01:35<00:00, 17.07s/it
Epoch: 0
--------
Train loss : 0.33213 | Train accuracy : 0.89%

Test loss: 0.07257 | Test accuracy: 0.98%

Epoch: 1
--------
Train loss : 0.07942 | Train accuracy : 0.98%

Test loss: 0.06755 | Test accuracy: 0.98%

Epoch: 2
--------
Train loss : 0.06264 | Train accuracy : 0.98%

Test loss: 0.05406 | Test accuracy: 0.98%

Epoch: 3
--------
Train loss : 0.05373 | Train accuracy : 0.98%

Test loss: 0.06162 | Test accuracy: 0.98%

Epoch: 4
--------
Train loss : 0.04864 | Train accuracy : 0.98%

Test loss: 0.04590 | Test accuracy: 0.98%
```

5. **Image with True label and predicted label:**



CONFUSION MATRIX: