

圖像分類：手寫數字識別

題目描述：

你將使用 PyTorch 來構建一個卷積神經網絡 (CNN) 模型，用於識別手寫數字，請按照以下步驟完成任務。

數據集：

- 訓練集圖像數據：`train-images.pt` (shape: [60000, 28, 28])
- 訓練集標籤數據：`train-labels.csv`
- 測試集圖像數據：`train-images.pt` (shape: [10000, 28, 28])

小題 1：數據讀取與預處理 (15 分)

1. 讀取數據集圖片和標籤 (5分)
2. 將數據集按 **8:2** 的比例劃分為訓練集和驗證集。(5分)
3. 將圖像數據歸一化並轉換為 Tensor。(5分)

```
1  import torch
2  import torch.nn as nn
3  import pandas as pd
4
5  # TODO: 讀取數據集，'train-images.pt' 和 'train-labels.csv'
6  images_raw = ...
7  labels_raw = ...
8
9  # TODO: 歸一化數據集並轉換為 torch.Tensor
10 images = ...
11 labels = ...
12
13 # TODO: 創建數據集，並按照 8:2 劃分成訓練集和驗證集
14 dataset = ...
15 train_dataset, val_dataset = ...
16
17 train_loader = DataLoader(train_dataset, batch_size=128, shuffle=True)
18 val_loader = DataLoader(val_dataset, batch_size=128)
```

小題 2：構建 CNN 模型 (15 分)

定義一個簡單的 CNN 模型，滿足以下條件：

1. 至少包含兩個卷積層 (`nn.Conv2d`) (2分)
2. 至少包含兩個池化層 (`nn.MaxPool2d`) (2分)
3. 使用激活函數 (`nn.ReLU`, `nn.Sigmoid`, `nn.Tanh`, `nn.ELU`, `nn.LeakyReLU`, `nn.PreLU` 等) (2分)
4. 定義輸出層，輸出一個 10 維向量 (對應 10 個類別) (2分)
5. 定義前向傳播，將以上所有網絡層連接 (7分)

```
1 class CNN(nn.Module):
2     def __init__(self):
3         super(CNN, self).__init__()
4         # TODO: 定義網絡層
5         ...
6     def forward(self, x):
7         # TODO: 定義前向傳播
8         ...
9
10 device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
11 model = CNN().to(device)
12 print(model)
```

小題 3：訓練模型 (20 分)

1. 選擇損失函數 (`MSELoss`, `CrossEntropyLoss` 等) (2分)
2. 選擇優化器 (`SGD`, `Adam` 等)，並設置學習率。(3分)
3. 模型訓練至少 5 個 epoch，並在每個 epoch 結束時分別打印訓練集和驗證集的損失函數和準確率。(15分)

```
1 # TODO: 定義損失函數和優化器
2 criterion = ...
3 optimizer = ...
4 for epoch in range(5):
5     # TODO: 訓練循環
6     model.train()
7     for batch_idx, (images, labels) in enumerate(train_loader):
8         ...
9         if batch_idx % 50 == 0:
10             print(f"Epoch {epoch+1}, Batch {batch_idx}, Train Loss: {train_loss:.4f},
11                 Train Accuracy: {train_acc * 100:.2f}%")
12
13     # TODO: 驗證循環
14     model.eval()
15     with torch.no_grad():
16         for images, labels in val_loader:
17             ...
18             print(f"Epoch {epoch+1}, Val Loss: {val_loss}, Val Accuracy: {val_acc}")
```

小題 4: 準確率得分 (50分)

根據模型在測試集 (賽後測試) 中的準確率評分

準確率區間 (%)	得分範圍	得分計算公式
90 - 100	30 - 50	$50 - 2 * (100 - x)$
80 - 90	20 - 30	$30 - (90 - x)$
50 - 80	0 - 20	$20 * (x - 50) / 30$
<50	0	0

可以使用以下代碼生成 `submission.csv` 並提交

```
1 import torch
2 import pandas as pd
3
4 test_images = torch.load('test_images.pt')
5 # TODO: 按照前面的方法歸一化
6 test_images = ...
7
8 model.eval()
9 with torch.no_grad():
10     test_images = test_images.to(device)
11     outputs = model(test_images)
12     predictions = outputs.argmax(dim=1)
13
14 df_test = pd.DataFrame({"label": predictions.cpu().numpy()})
15 df_test.to_csv("submission.csv", index_label="id")
```