EXERCISE

NEURAL NETWORKS

Requirements:

- Build Multi Layer Perceptron (MLP) and LeNet-5 model.

- Experiment on Fashion-MNIST and CIFAR-10 dataset.

- The structure of MLP model for each specific dataset:

Fashion-MNIST		
Layer	#Nodes	Activation
Input layer	782	ReLU
Hidden layer 1	100	ReLU
Hidden layer 2	100	ReLU
Hidden layer 3	50	ReLU
Output layer	25	Softmax(dim = 1)

CIFAR-10		
Layer	#Nodes	Activation
Input layer	3072 (= 3 * 32 * 32)	ReLU
Hidden layer 1	1024	ReLU
Hidden layer 2	512	ReLU
Hidden layer 3	256	ReLU
Output layer	64	Softmax(dim = 1)

- Hyperparameters for training model (both models):

  + random_seed = 1 + batch_size = 64

  + n_epochs = 50 + learning_rate = 0.001

- Optimizer: Adam

- Loss function: Cross Entropy

- Performance metric: Accuarcy

Tutorials:

- Use ‘torchvision.dataset.FashionMNIST’ to load Fashion-MNIST dataset and ‘torchvision.dataset.CIFAR10’ to load CIFAR-10 dataset.

- For Fashion-MNIST dataset, it is as same as the MNIST dataset. Therefore, we experiment both models on Fashion-MNIST dataset as the same way we experiment on the MNIST dataset.

- For CIFAR-10 dataset, each datapoint is a color image with 32 x 32 size. Thus, we have to change some hyperparameter values of model to fit this dataset:

  + For MLP model, we change the value of ‘in_channels’ of ‘input’ layer to 3 * 32 * 32 = 3072 (color channel * width * height).

  + For LeNet-5 model, we change the value of ‘in_channels’ of ‘input’ layer to 3 (because the input is a color image) and we don’t need to reshape the image because the required input of LeNet-5 is 32 x 32.

- To someone wants to show an image in CIFAR-10 dataset, please run below lines of codes

```
>> import matplotlib.pyplot as plt
   image, label = train_dataset[0]
   plt.imshow(image.permute((1, 2, 0)))
   print(classes[label])
```