

Artificial Intelligence Lab Work (4)

問題 1 第 10 回の講義資料に基づいて MNIST の学習・推論プログラムを実装し(スライドにかかれて
いるプログラムを写す)、そのプログラム(.py)とコンソールに表示される実行結果(学習中の各エポック
での損失とテストデータに対する精度)を word ファイルに貼り付けて提出せよ。

(2 点)

問題 2 問題 1 で作成したプログラムを書き換えて、CIFAR-10 という画像認識のデータセットに対し
て、学習と推論を行い、そのプログラム(.py)とコンソールに表示される実行結果(学習中の各エポックで
の損失とテストデータに対する精度)を word ファイルに貼り付けて提出せよ。CIFAR-10 は 10 クラスの
画像分類データであり、

```
train_dataset = tv.datasets.CIFAR10(root="./", train=True,  
                                     transform=tv.transforms.ToTensor(),  
                                     download=True)  
test_dataset = tv.datasets.CIFAR10(root="./", train=False,  
                                    transform=tv.transforms.ToTensor(),  
                                    download=True)
```

を実行することにより、データが得られる。各画像は 3 色×32pixel×32pixel となっていることに注意せ
よ。

(2 点)

問題 3 問題 2 で作成したプログラムを書き換えて、中間層を一つ畳み込み層にした NN を作成し、学
習と推論を行い、そのプログラム(.py)とコンソールに表示される実行結果(学習中の各エポックでの損失
とテストデータに対する精度)を word ファイルに貼り付けて提出せよ。畳み込み層は、

```
nn.Conv2d(in_channel, out_channel, filtersize)
```

により得られる。in_channel は入力のチャンネル数、out_channel は出力のチャンネル数、filtersize はフィル
ターのサイズである。

(2 点)

Artificial Intelligence Lab Work (4)

Q1 Implement the MNIST learning and inference program by following the 10th lecture's slides (copy the program on the slide), and submit the program (.py) and the execution results, i.e., loss at each epoch during training and accuracy against test data, displayed on the console in a word file.

(2 points)

Q2 Rewrite the program you wrote in Q1 to train and infer on the image recognition dataset CIFAR-10, and submit the program (.py) and the execution results, i.e., loss at each epoch during training and accuracy against test data, displayed on the console in a word file.

CIFAR-10 is a 10-class image classification data, and can be downloaded by the following program.

```
train_dataset = tv.datasets.CIFAR10(root=".", train=True,
                                     transform=tv.transforms.ToTensor(),
                                     download=True)
test_dataset = tv.datasets.CIFAR10(root=".", train=False,
                                    transform=tv.transforms.ToTensor(),
                                    download=True)
```

Note that each image has 3 colors \times 32 pixels \times 32 pixels.

(2 points)

Q3 Rewrite the program you wrote in Q2 to create an NN with one intermediate layer as a convolutional layer, and submit the program (.py) and the execution results, i.e., loss at each epoch during training and accuracy against test data, displayed on the console in a word file. The convolutional layer can be obtained by the following program.

```
nn.Conv2d(in_channel, out_channel, filtersize)
```

where in_channel is the number of input channels, out_channel is the number of output channels, and filtersize is the size of the filter.

(2 points)