

Homework 4

Haocong Wang

I chose PyTorch as my framework and built three networks based on the requirement.

The following is the result of the classic LeNet5:

```
Train Epoch: 50 [43520/50000 (87%)] Loss: 1.108510
Train Epoch: 50 [44800/50000 (90%)] Loss: 1.071049
Train Epoch: 50 [46080/50000 (92%)] Loss: 1.282843
Train Epoch: 50 [47360/50000 (95%)] Loss: 1.376124
Train Epoch: 50 [48640/50000 (97%)] Loss: 1.109919
Train Epoch: 50 [31200/50000 (100%)] Loss: 1.331077

Test set: Average loss: 1.1858, Accuracy: 5946/10000 (59%)

Traning and Testing total excution time is: 1063.574051618576 seconds
```

The next one is the result of the LeNet5 with one additional dropout layer:

```
Train Epoch: 50 [43520/50000 (87%)] Loss: 1.358558
Train Epoch: 50 [44800/50000 (90%)] Loss: 1.544693
Train Epoch: 50 [46080/50000 (92%)] Loss: 1.494913
Train Epoch: 50 [47360/50000 (95%)] Loss: 1.299029
Train Epoch: 50 [48640/50000 (97%)] Loss: 1.285614
Train Epoch: 50 [31200/50000 (100%)] Loss: 1.438040

Test set: Average loss: 1.1824, Accuracy: 5988/10000 (60%)

Traning and Testing total excution time is: 1049.5816481113434 seconds
```

The last one is the result of the LeNet5 with one additional batch normalization:

```
Train Epoch: 50 [43520/50000 (87%)] Loss: 0.785285
Train Epoch: 50 [44800/50000 (90%)] Loss: 0.886248
Train Epoch: 50 [46080/50000 (92%)] Loss: 0.776391
Train Epoch: 50 [47360/50000 (95%)] Loss: 0.944488
Train Epoch: 50 [48640/50000 (97%)] Loss: 0.821841
Train Epoch: 50 [31200/50000 (100%)] Loss: 1.230140

Test set: Average loss: 0.8938, Accuracy: 6867/10000 (69%)

Traning and Testing total excution time is: 1072.707855463028 seconds
```