Pattern Recognition Group Exercise 2c

Davud Bucher, Timo Bürk, Félicien Hêche, Aleksandar Lazic, Zahkar Tymchenko

April 19, 2020

1 Incomplete CNN Implementation

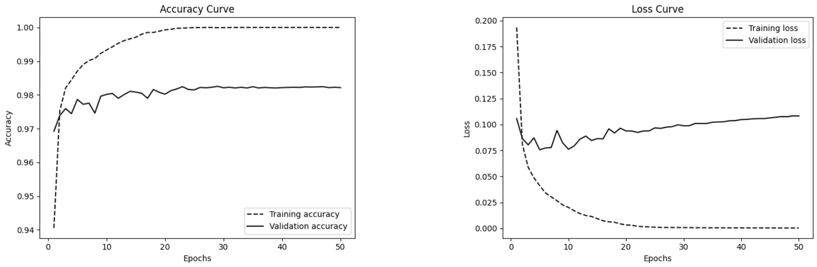
The input are picture of 28 by 28 grey scale pixels, thus the $\mathit{expected\_input\_size} = (28, 28)$ and the $\mathit{in\_channels} = 1$. We then find the number of out put channel and kernel size by trying combination of value that satisfy the formula

$$\mathit{output\_channels} * [(i + 2p - \mathit{kernel\_size})/s]^2 \overset{!}{=} 1536$$

with $i = 28, p = 0, s = 3$. We find $\mathit{output\_channels} = 24$ and $\mathit{kernel\_size} = 7$. Finally the their are ten labels 0,...,9.

2 CNN Tasks

We implement our CNN using the provided model_task2c.py and pyTorch. We use the value detailed above.
From the learning rates $[0.001, 0.0025, 0.005, 0.0075, 0.01, 0.025, 0.05, 0.075, 0.1]$ we find 0.01 to work the best.
We then train our CNN and plot the accuracy and loss on the train and validation set for each epoch.



(a) Accuracy

(b) Loss

Figure 1: CNN Performance During Training

As we can see in [Figure 1a] the accuracy on the training set goes to 100% as we let the cnn train, but the accuracy on the validation do not increase after 20

epochs and stays at about 98%. Now in [Figure 1b] we can see that the loss on the training set goes to 0 as we let the cnn train. On the validation set we can see something interesting, the loss decrease on the first few iteration, then stays about the same for the epochs 10 to 20, and then increase. This can suggest some overfitting problems. Thus an optimal number of training iteration is somewhere between 10 and 20 epochs, as the accuracy on the validation set increase until the 20 epochs, we choose 20 as the optimal training iteration number.
We then randomly initialze our model multiple time and select the best one based on its accuracy on the validation training set. We test this model on the test set, giving us the following results: 0.9851 accuracy and 0.0625 Loss