Introduction to Machine Learning (Spring 2019)

Homework #5 (50 Pts, June 5)

Student ID _____

Name _____

Instruction: We provide all codes and datasets in Python. Please write your code to complete Convolutional Neural Network Classifier. Compress 'Answer.py' & your report ONLY and submit with the filename 'HW5_STUDENT_ID.zip'.

(1) [30 pts] Implement CNN Classifier in 'Answer.py' with the loss function as follows:

$$L = \frac{1}{N}\sum_{i=1}^{N} L_i,$$

$$L_i = -\sum_{j=1}^{C} y_j log p_j,$$

where N is the number of (batch) data, C is the number of classes.

(a) **[Convolution 2D]** Implement convolution function in 'Answer.py' ('convolution2d').

(b) **[ReLU]** Implement ReLU activation in 'Answer.py' ('ReLU').

(c) **[Convolution Layer]** Implement a convolution layer in 'Answer.py' ('ConvolutionLayer').

(d) **[Max-Pooling Layer]** Implement a max-pooling layer in 'Answer.py' ('MaxPoolingLayer').

(e) **[FC Layer & Softmax]** Implement a FC, softmax layer in 'Answer.py' ('FCLayer', 'SoftmaxLayer').

Answer: Fill your code here. You also have to submit your code to i-campus.

NOTE 1: **You should write your codes in 'EDIT HERE' signs.** It is not recommended to edit other parts. Once you complete your implementation, run the check codes ('Checker.py') to check if it is done correctly.

NOTE 2: **Read the instructions in template codes VERY CAREFULLY.** Funcionality and input, output shape of any function must be the same as what is written.

(2) [20 Pts] Experiment results

(a) you are given a small MNIST dataset with 5 labels (0, 1, 2, 3, 4), which originally has 10 labels. Given CNN architecture and hyperparameters as below, build the classifier and adjust hyperparameters to achieve best test accuracy. (Your best accuracy should be at least 0.8 if the model is trained correctly.)

Answer: Fill the blank in the table. Show the plot of training & test accuracy with a brief explanation.

[CNN Architecture]

Layer name	Configuration
Conv - 1	Out Channel = 8, Kernel size = 3 Stride = 1, Pad = 1
ReLU - 1	-
Conv – 2	Out Channel = 8, Kernel size = 3 Stride = 1, Pad = 1
ReLU - 2	-
Max-pool - 1	Kernel size = 2, stride = 2
FC – 1	Input dim = 1568, Output dim = 500
FC - 2	Input dim = 500, Output dim = 5
Softmax Layer	-

[Results]

Epochs	Learning rate	Best Acc.	Best Epoch.

Plot Sample (Values are not correct. Delete when you submit).