Assignment 1
CSL7590: Deep Learning
AY 2023-24, Semester – II
Due on: 24-01-2024

M.M: 100

General Instructions:

1. Clearly mention the assumptions you have made, if any.
2. Clearly report any resources you have used while attempting the assignment.
3. Any submission received in another format or after the deadline will not be evaluated.
4. Make sure to add references to the resources that you have used while attempting the assignment.
5. Plagiarism of any kind will not be tolerated and will result in zero marks.
6. Select your dataset correctly. If found otherwise, your assignment will not be evaluated.

Submission Guidelines:

1. Prepare a Python code file for the task and name it as **YourRollNo.py.** There should be **one and only one .py file**. No need to prepare a separate .py file per subtask. **The .py files must not be named like <roll no>_task1(1).py**
2.Submit a single report depicting methods, results, and observations. There is no need to add theory behind the concepts. Preparing a report is mandatory; failing it will lead to non-evaluation of the assignment.
3. Name your report as **YourRollNo.pdf.** Also, **provide your colab file link in the report.** Make sure that the file is sharable.
4. There is **no need to make a zip file.** Just upload both the codes and a report directly on the google-classroom, that is, submission will contain {YourRollNo.py and YourRollNo.pdf}. **Do not upload files in any other format.**
5. Do not download the .ipynb file, rename it as .py, and upload it. .ipynb files are not exactly in a readable form, so uploading it will only result in you receiving 0 marks for the same. You have an option to download a .py file in google colab. Use it to get the .py format.
6. Do not copy-paste code or screenshots, etc. in the report. The report should look like a technical document, containing plots, tables, etc. whenever necessary.
7. **Adhere to the instructions given, failing them may result in a penalty.**

Objective:

In this assignment, you are required to implement a convolution neural network from scratch in Python using the Pytorch framework. By the end of this assignment, you should have a working neural network that can be trained on a simple dataset for multi-class classification.

Dataset:

Get the dataset from here. The dataset consists of handwritten digits; 60k images for training and 10k images for testing.

Network Architecture:

1. The network should contain 3 fully convolutional layers. Excluding fully connected output layer. Set the network architecture as:
    a. Convolution layer1: kernel size=7*7; Maxpool;Stride=1; output channels=16
    b. Convolution layer2: kernel size=5*5; Maxpool; Stride=1; output channels=8
    c. Convolution layer3: kernel size=3*3; Average pooling; Stride=2; output channels=4
    d. Output layer: set to the size of the #classes
2. Use zero padding to preserve the input image dimension.
3. Use standard train-test splits.
4. Let A be the last three digits of the roll number:
    a. if A is divisible by 2, then set batch size = 32;
    b. if A is divisible by 3, then set batch size = 16;
    c. else, batch size = 20

    For example, if your roll number is M23CS021, your batch size should be 16 as 21%3=0.
5. Set 'ReLU' as the activation function for convolution layers and 'softmax' for the output layer.
6. Use Adam optimizer; loss function as cross-entropy.
7. Train for 10 epochs. Plot accuracy and loss per epoch. If you get low accuracy, train for more epochs.
8. Prepare a Confusion matrix for the **test** set for all the combinations of the network. You may use an in-built function for this purpose.
9. Report total trainable and non-trainable parameters.

Experiment 1: **[50 marks]**

Implement a Convolutional Neural Network to solve a 10-class classification problem using Pytorch as described above.

Experiment 2: **[30 marks]**

Combine the data (digit images) to create the following classes:

Class 1: {0, 6}
Class 2: {1, 7}
Class 3: {2, 3, 8, 5}
Class 4: {4, 9}

Use the CNN model implemented in Experiment 1 to solve a 4-class classification problem.

Report: **[10 marks]**

Prepare a detailed report containing all the results, graphs, plots, methodology, and observations.

Bonus: Use techniques to improve performance and avoid overfitting, if it occurs. [10 marks]

Note: Do not use in-built functions unless mentioned.