APL 745: Deep Learning for Mechanics

Lab – 2

To be submitted by 3rd March midnight

The objective is to develop (step by step) a convolutional neural network to perform classification problem.

Load the following packages for this problem:

- PyTorch is a deep learning package. (other deep learning packages can also be used).
- Numpy is the fundamental package for scientific computing with Python.
- H5py is a common package to interact with a dataset that is stored on an H5 file
- Matplotlib is a famous library to plot graphs in Python.
- Sklearn is a commonly used Python package for basic data pre-processing.

Step 1: Load the data (problem1.hdf5) using H5py and Numpy. The dataset (MNIST dataset) consists of 28×28 pixel grayscale images of handwritten single digits from 0 to 9 with the corresponding true labels.

Step 2: Define the following model structure with different Python files as follows:

- Model-configuration – 1: Build 2 convolutional and 2 linear layers as follows:

  Input → Conv1 (with output channels = 16) → Conv2 (with output channels = 32) → Maxpool → Flatten → linear1 (512 hidden units) → linear 2 (128 hidden units) → output.

- Model-configuration – 2: Build 2 convolutional and 2 linear layers as follows with ReLU activation and dropout as follows:

  Input → Conv1 (with output channels = 16) → ReLU → Conv2 (with output channels = 32) → ReLU → Maxpool → dropout (0.5) → Flatten → linear1 (512 hidden units) → ReLU → dropout (0.5) → linear 2 (128 hidden units) → output.

- For all conv layers, use kernel of size 5x5 with no padding and stride = 1
- For all max pool layers, use window of size 2x2 with no padding and stride = 2

- For all the above configurations, use a suitable output activation function for this classification problem.

Step 3: Train the NN model using ADAM optimizer.

Step 4: Test the NN model using the test data that was unseen during training. Make sure that the gradient calculation is disabled during this inference stage

Write a computer code to answer the following questions related to Step-1:

(A) Split the dataset into two parts as training data and testing data with the ratio 70 : 30.
- Print the input and the output shape of the training and the testing dataset.
- Plot first 5 test images and provide the true label as the caption for each test image.

(B) Create an iterable DataLoaders that we can feed into a neural network for training and testing. Use batch size as 64.

Write a computer code to answer the following questions related to Step-2 and -3:

(C) Use a suitable loss function for training the NN model.

(D) Train all the above model-configurations with the following learning rate and weight decay training-configurations:
- Training-configuration-1: learning rate: 5×10^{-2}, weight decay: 1×10^{-3}
- Training-configuration-2: learning rate: 1×10^{-4}, weight decay: 1×10^{-5}

(E) Plot the following separately for the training data and test data for all the above configurations:
- Epoch vs Training loss
- Epoch vs Testing loss
- Epoch vs Accuracy

(F) Plot the first 5 ground truth test data, the ground truth label, and the predicted labels as the caption.

(G) Print the number of parameters for all the above model-configurations.

(H) Compare the results obtained with those obtained in Lab-1. Choose (training config 1 + model config 2 + batch size 64) for CNN and (training config 1 + model config 4 + batch size 64) for FNN (from Lab 1).

(I) Comment on your observations along with the computational time (wall clock time in seconds) for the above comparison.