APL 745: Deep learning for mechanics

Lab – 1

To be submitted by midnight of 2nd Feb, 2022

In this problem, the objective is to develop (step-by-step) a neural network with fully connected layers 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 MNIST dataset (MNISTdata.hdf5) using H5py and Numpy. The 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 a 1-hidden layer network with 50 hidden units.
- Model-configuration – 2: Build a 2-hidden layer network with 400 and 200 hidden units for the first layer and second layer, respectively.
- Model-configuration – 3: Build a 2-hidden layer network with 400 units in the first hidden layer and 200 units in the second hidden layer, respectively. Use the ReLU activation between the input and hidden layer-1, and between the hidden layer-1 and hidden layer-2.
- Model-configuration – 4: Build 2-hidden layer network with 400 and 200 hidden units respectively with following configuration:
  Input →linear1→batch normalization→dropout→ReLU→linear2→output
- For all the above configurations, use a suitable output activation function for this classification problem.

Step 3: Train the NN model using the ADAM optimization method.

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 sizes 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) Flatten the training and the testing dataset by reshaping the images into a single vector for each data. Print the input and the output shape of the training and the testing dataset.

(C) Create DataLoaders that can be iteratively fed to 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:

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

(E) 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^{-3}, weight decay: 1×10^{-4}
- Training-configuration-3: learning rate: 1×10^{-4}, weight decay: 1×10^{-5}

(F) For all the model-configurations combined with training-configuration-2, train the NN model with mini-batch sizes of 64, 256, 512.

(G) 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

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

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

(J) Provide your observations along with the computational time (wall clock time in seconds) for the above configurations.