AI502: Deep Learning (Spring 2023)

Programming Assignment 1

Due: April 9 (Sun), 11:59 pm

1 Overview

In this programming assignment, you will implement a fully connected neural network (FCN) and a convolutional neural network (CNN) for the CIFAR-10 image classification dataset. The purpose of this assignment is to give you hands-on experience with building and training neural networks using PyTorch. Specifically, the goal is to classify images into different categories, with the model outputting a probability for each class given a $32 \times 32 \times 3$ input image.

2 Fully Connected Neural Network

First, you will implement a fully connected neural network, which is composed of multiple feedforward layers, where each layer is followed by an activation function as shown in the equation below. Note that the image must be flattened before it is fed to the first layer.

$$h_l(x) = \sigma(\mathbf{A}h_{l-1}(x) + b) \tag{1}$$

Please complete FCNN class in the provided skeleton code.

3 Convolutional Neural Network

Here you will implement a convolutional neural network to classify CIFAR-10 images using torch.nn.Conv2d. Similar to FCNN, each layer is followed by an activation function. Note that the last layers can be composed of FCNN for classification.

- **in_channels**: Number of channels in the input image
- **out_channels**: Number of channels produced by the convolution
- **kernel_size**: Size of the convolving kernel
- **stride**: Stride of the convolution
- **padding**: Padding added to all four sides of the input

Please complete CNN class in the provided code.

4 Experiments with various settings

Try to train the model with various settings with choices below.

- Model architecture: Number of layers, size of hidden dimensions, activation functions, etc.
- Optimizer: Adam, RMSProp, SGD, etc.
- Number of training epochs
- Learning rate
- Batch size
- etc.

5 Report

The report should contain the followings.

- **Model architecture**: Try with various structures and report the best model architecture with its loss and accuracy.
- **Comparison between FCNN and CNN**: Compare FCNN and CNN in terms of loss, accuracy, number of parameters, etc.
- **Effects of hyperparameters**: Try different hyperparameter sets (it is recommended to change at least three different types of hyperparameters) and report their effects to training. For this you may need to do grid search over hyperparmeters. Try to provide a result table or plot to verify your opinion.

6 How to submit

1. Download and fill in the skeleton codes. You can find the skeleton code on KLMS or a colab link. The data can be also downloaded from this link. Refer to following documentation if you are not familiar with PyTorch library, or ask TAs for help.
2. Experiment with various settings and write a report.
3. Modify the name of files and compress them into a zip file.
   - code: **PA1_{student_ID}_{name}.ipynb**
   - report: **a PDF format**, with a filename as **PA1_{student_ID}_{name}.pdf**
   - zip: **{student_ID}_{name}.zip**

The report should be in at most 2 pages.