ENPM 808Y: Neural Networks, Spring 2020

Assignment 2: Convolutional Neural Network

The aim of this project is to implement a Convolutional Neural Network to achieve hand-written digit recognition and image classification. Analyze the performance and the strengths and weaknesses/limitations of the classifier. The whole pipeline is described in the following section.

1. Classifier

Build a Convolutional Neural Network. Train it on the MNIST and CIFAR-10 training set and test it on the testing set. You can design your architecture or use the architecture introduced in LeCun's paper [1]. You can use any of the following toolboxes:

Python Tools: Pytorch, Tensorflow, Keras

Note that MNIST classification models (with pre-trained weights) are included in most DL framework example. You may test these readily available codes, but you should provide your own implementation under the framework you choose, and compare with the pretrained models.

2. Datasets

MNIST:

You can download MNIST from http://yann.lecun.com/exdb/mnist/. The description of Dataset is also on the website. Basically it has a training set with 60000 28 x 28 grayscale images of handwritten digits (10 classes) and a testing set with 10000 images.

Best classifier: Regularization of Neural Networks using DropConnect TEST ERROR RATE : 0.21% ICML 2013 (http://yann.lecun.com/exdb/publis/pdf/wan-icml-13.pdf)

CIFAR-10:

You can download the data from CIFAR-10 webpage (http://www.cs.toronto.edu/~kriz/cifar.html). The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The dataset consists of 10 classes: Airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck.

Best classifier: Network In Network, Accuracy : 91.2%, ICLR 2014 (https://openreview.net/forum?id=ylE6yojDR5yqX)

1) It will be up to you to design the best network as possible as you can. Show the training and generalization performance for each class data and also for the total data using the table below, using top-1 error rate. Also, repeat using the top-3 error rate.

											(%)
Average precision	Airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck	**Mean**
training											
test											

2) Show the confusion matrix as given by Fig. 5 of "Multi-column Deep Neural Networks for Image Classification" by Ciresan et al.

3. Report

The report should include details on :

- What you learned through this assignment.
- Implementation of your convolutional neural network architecture.
- Confusion matrix
- Analyze and report hyperparameters obtained. (optimizer, batch size, epochs, layers etc)
- Training, validation, test curves and classification result.
- Best hyperparameters and train-test split obtained and your interpretation of these results.

Please ensure that you analyze and compare your results for various parameters and draw conclusions.

4. Bonus

The goal of this bonus is for you to push your best as much as possible on the Image Classification problem. Your bonus is determined by your test error rate. You will get extra credits if you outperform your peers.

5. Submission Guidelines

The submission should include the following:

- Code
- Report (in PDF format). The results must be presented in a compact form and to the point in your report description in order to gain more credit.
- Readme.txt with instructions to run your code.

The file should be named DirectoryID_HW2.zip and submitted to ELMS/Canvas.

References

[1] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document
recognition. Proceedings of the IEEE, Nov 1998.
[2] Tensorflow models: https://github.com/tensorflow/models
[3] Pytorch models: https://pytorch.org/docs/stable/torchvision/models.html

NOTE: "If you have any questions regarding the assignment (HW2), then try to ask in the "Discussions" so that everyone can benefit."