CS/ECE 8690 Computer Vision

Homework 3B – Image Classification [60 pts]

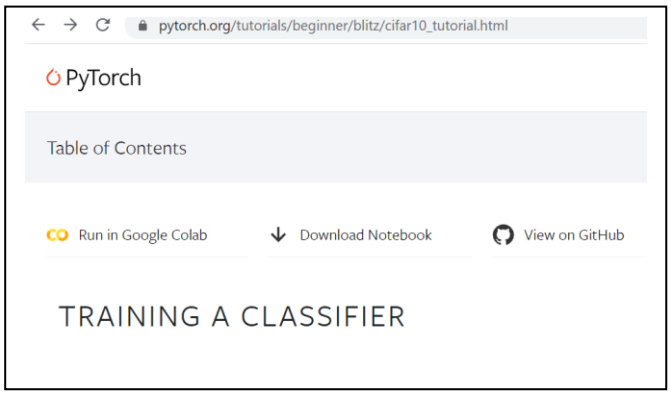
Out: Tuesday Mar 14 **Spring 2023**
Due: Tuesday Mar 21

The goal of this assignment is to create, train, and test a convolutional neural network for image classification.

You will modify the image classification code given at:
https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html

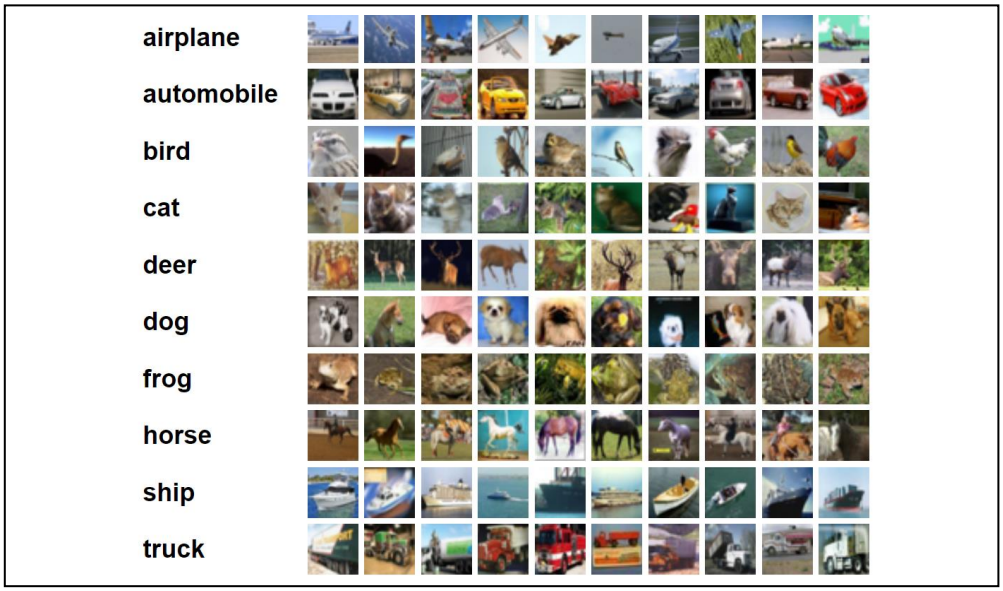
The tutorial includes code for the following steps.

1. Load and normalize the CIFAR10 training and test datasets using torchvision
2. Define a Convolutional Neural Network
3. Define a loss function
4. Train the network on the training data
5. Test the network on the test data



You can download the notebook and run on your computer or use Google Colab (https://colab.research.google.com/).

DATASET: We will be using the CIFAR10 dataset (https://www.cs.toronto.edu/~kriz/cifar.html) , consisting of 60,000 images of 10 common classes. Each image is of size 32 x 32 x 3. There are 50,000 training images and 10,000 test images. CIFAR10 is one of the built-in datasets Torchvision provides (https://pytorch.org/vision/stable/datasets.html).



ASSIGNMENT

I. **Network-1:** Train, test, and evaluate the convolutional neural network (CNN) given at https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html
   a. Train the network for 4 epochs
   b. Report overall accuracy (# of examples correctly classified/# of examples)
   c. Report classification accuracy per class
   d. Report confusion matrix (A 10 x 10 table, where the cell at row i and column j reports the fraction of times an example of class i was labeled by your model as class j. Please label the rows/columns by the object class name, not indices.

II. **Network-2:** Modify the above tutorial code to create your own network as described below

Layer	Kernel Size	Input Channels	Output Channels
Conv + RELU	5x5	3	6
MaxPool	2x2		
Conv + RELU	5x5	6	16
MaxPool	2x2		
Conv+RELU	5x5 (padding=2)	16	32
Fully Connected			120
Fully Connected			10

   a. Train the network for 4 epochs
   b. Report overall accuracy (# of examples correctly classified/# of examples)
   c. Report classification accuracy per class
   d. Report confusion matrix (A 10 x 10 table, where the cell at row i and column j reports the fraction of times an example of class i was labeled by your model as class j. Please label the rows/columns by the object class name, not indices.

III. **Network-3:** Modify the network above to perform 4-class image classification where the classes are {'cat','car','frog','other'}. Note: Do not regroup the classification outputs, create a 4-class classification network, train and test the new 4-class network.
   a. Train the network for 4 epochs
   b. Report overall accuracy (# of examples correctly classified/# of examples)
   c. Report classification accuracy per class
   d. Report confusion matrix (A 4 x 4 table, where the cell at row i and column j reports the fraction of times an example of class i was labeled by your model as class j. Please label the rows/columns by the object class name, not indices.

REPORT:
Your report should include.

1. Accuracy table for 10-class classification (Network 1 and Network 2)
2. Confusion matrices for 10-class classification (Network 1 and Network 2)
3. Accuracy table for 4-class classification (Network 3)
4. Confusion matrices for 4-class classification (Network 3)
5. **Q & A:** 10-class classification: for the class on which your model (Network 2) has the worst accuracy, what is the other class it is most confused with?
6. **Visualization:** Show 5 test images that your model (Network 2) confused between these classes and comment on what factors may have caused the poor performance.

Accuracy Tables

	Net1	Net2
Plane		
Car		
Bird		
Cat		
Deer		
Dog		
Frog		
Horse		
Ship		
Truck		
AVERAGE		
Running time		

	Net1	Net2
Car		
Cat		
Frog		
Other		

Confusion Matrices

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
Plane										
Car										
Bird										
Cat										
Deer										
Dog										
Frog										
Horse										
Ship										
Truck										

	Car	Cat	Frog	Other
Car				
Cat				
Frog				
Other				

Submission instructions: Your submission should **include a separate report** (including above-described results) and associated programs (as separate files).

References:

[1] Lec11 and Lec12 slides on Canvas
[2] https://pytorch.org/vision/main/index.html