Homework #1 – Choosing a Deep Learning Framework and Setting up an Environment
CAP 5619, Deep & Reinforcement Learning (Spring 2021), Department of Computer Science, Florida State University

---

Points: 45
Due: Beginning of the class (6:35pm) on Monday, January 24, 2022

Submission: You need to submit electronically via Canvas by uploading a) a pdf file (named "**hw1-Firstname-Lastname.pdf**") for your answers to the questions, and b) the program(s) for Problem 2 (named as "**hw1-prog-Firstname-Lastname.zip**"); if there are multiple files, please zip them as a single archive. Here replace "Firstname" using your first name and replace "Lastname" using your last name in the file names

The main purpose of this assignment is to let you be familiar and become comfortable with the deep learning framework of your choice.

Problem 1 (20 points) Find an example of a (deep) neural network (either fully connected or convolutional) using MNIST in the deep learning framework of your choice. If you use Keras, you can find a number of examples at https://keras.io/examples/ including https://github.com/keras-team/keras-io/blob/master/examples/vision/mnist_convnet.py (convolutional neural network for MNIST); you can replace the convolutional neural network by a multi-layer perceptron (which is fully connected) using the example from https://www.cs.fsu.edu/~liux/courses/deepRL/examples/mnist_mlp.py. You can also find a pytorch example from https://github.com/pytorch/examples/tree/master/mnist. For Tensorflow examples, please see https://www.tensorflow.org/tutorials. After you train the network in an environment you set up, answer the following questions.

(1) Briefly describe the network architecture you have, including how many layers, what kinds of layers (including activation functions and the number of trainable parameters).
(2) Generate and include the loss and recognition accuracy curves with respect to the number of epochs and report the performance on the training set and test set (which is actually a validation set in many implementations).
(3) Try to improve the performance by adding an additional layer in the model you use. Describe your attempts and results.

Problem 2 (10 points) Now modify the program you have for Problem 1 with the best performance by assigning random labels (between 0 and 9) to the samples in the training set. After you train the network on the modified dataset, answer the following questions. (Hint: reading the paper in the next question first may be helpful.)

(1) Report the performance on both the training set and test set by plotting the loss and recognition accuracy with respect to the number of epochs.
(2) Summarize the main performance difference between the two versions and try to explain why. (Hint: has the gap between the accuracy on the training set and the one on the test set increased significantly?)

Problem 3 (15 points) Read the paper "understanding deep learning requires rethinking generalization" (available from https://arxiv.org/pdf/1611.03530). Explain the challenges of achieving good generalization and some general strategies to improve generalization based on your experience with Problems 1 and 2 and the experiments in the paper. In particular, address the issue whether one could establish a bound on the generalization error of deep neural networks in general. (Hint: if a deep neural network could memorize a training dataset with randomly assigned labels, is it possible for anyone to say anything about its generalization error in general?)

Further Information

The Computer Science Department has two GPU servers (aurora1.cs.fsu.edu and aurora2.cs.fsu.edu), each of which has a GeForce RTX 3090 GPU with 24 GB of GPU memory. See below for more information.

For this class and many deep learning projects that do not require very large datasets, a free Google colab account should be sufficient; see https://colab.research.google.com/?utm_source=scs-index# to learn more details.

```
liux@aurora1:~>nvidia-smi
Thu Dec 30 23:39:21 2021
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.32.03    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  GeForce RTX 3090    Off  | 00000000:01:00.0 Off |                  N/A |
|  0%   34C    P8    17W / 350W |    103MiB / 24245MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A      1970      G   /usr/lib/xorg/Xorg                 86MiB |
|    0   N/A  N/A      2633      G   /usr/bin/gnome-shell               14MiB |
+-----------------------------------------------------------------------------+
```

In general, for this class the primary goal is to gain systematic understanding of deep and reinforcement approaches and models, not running models with a very large number of parameters.