# Spring 2023 CS 747 Deep Learning
# Assignment 2

Due date: March 1, 2023, 11:59 pm

**Announcement**

- Honor Code: Everyone needs to submit their own solution individually for this assignment. When you submit your solution, it must be your own work. Any sources must be cited if it is not from you. Otherwise you may violate the GMU honor code and GMU CS honor code.
- Please note that this assignment requires using Pytorch, and it needs to run on GPUs. You are highly recommended to use Google Colab to finish this assignment, unless you have a local machine with powerful GPUs. Since the process of running one model is time-consuming, **please start the assignment early**.

## 1 Multi-label Image Classification (100 points)

In this part of the assignment you will implement a multi-label image classifier on the PASCAL VOC 2007 [1] dataset. You will design and train deep convolutional networks to predict a binary present/absent image-level label for each of the 20 PASCAL classes. This will help you gain experience with PyTorch. You will do the following:

### 1.1 Pre-defined Models (20 points)

This part is a warm-up practice of Pytorch, in which you can get a sense of how deep learning is implemented and what is the correspodning performance on the Pascal VOC dataset. The file `CS747_Assignment2_part1.ipynb` will guide you to start. Run the following experiments and report your results and observations.

- Train AlexNet [2] (Pytorch built-in) from scratch.
- Fine-tune AlexNet which is pretrained on ImageNet [3].
- Train a simple neural network (defined in `classifier.py`) from scratch.

[1] `http://host.robots.ox.ac.uk/pascal/VOC/voc2007/`
[2] `https://pytorch.org/hub/pytorch_vision_alexnet/`
[3] `http://www.image-net.org/`

### 1.2 Self-designed Models (80 points)

The file `CS747_Assignment2_part2.ipynb` will guide you to start. Design your own model in details to solve this multi-label classification task.

- No pre-trained model is allowed for this part.
- You can use concepts or ideas from existing models (e.g., VGG, ResNet, DesnetNet, etc.), but you cannot directly copy the existing models. Please include your own ideas reflected on the model design.
- You may want to start your design from either a simple network provided in `classifiers.py` or AlexNet.

## How to start

The jupyter notebooks `CS747_Assignment2_part1.ipynb` and `CS747_Assignment2_part2.ipynb` will guide you through all steps of the classifier. You will be required to use a GPU for this assignment. I suggest everyone to use Google Colab.

**Environment Setup (Local)** If you will be working on the assignment on a local machine then you will need a python environment set up with the appropriate packages. We suggest that you use Conda to manage python package dependencies (https://conda.io/docs/user-guide/getting-started.html).

Unless you have a machine with a GPU, running this assignment on your local machine will be very slow and is not recommended.

**Data Setup (Local)** Once you have downloaded the zip file, go to the Assignment3 folder and execute the download_data script provided:

```
./download_data.sh
```

**Data Setup (For Colab)** If you are using Google Colaboratory for this assignment you will need do some additional setup steps.

You will need to run the download_data.sh script inside colab cells and move all the data to a local directory in colab for fast access.

You will now need to open the assignment 2 ipython notebook file from your Google Drive folder in Colaboratory and run a few setup commands. Make sure to set the GPU as the hardware accelerator. To do this, on the top bar choose Edit→Notebook Settings→select 'GPU'. We have condensed all the important setup commands you need to run into an ipython notebook in the file `colab_setup.ipynb`.

**IPython** If you are using Colaboratory, you can open the ipython notebook directly in Colaboratory. If you are using a local machine, ensure that ipython is installed (`https://ipython.org/install.html`). You may then navigate the assignment directory in terminal and start a local ipython server using the jupyter notebook command.

## Submission Instructions

Please submit all your files on GMU Blackboard Portal before the due date.

- Upload your output file of **self-designed models (part 1.2)** to the Kaggle Competition for the Pascal Classifier [4]
- All of your code (python files and ipynb file) in a single ZIP file. Please include a `README` file about instructions for your submission. The filename should be `netid_assignment2_code.zip`.
- Your ipython notebook with output cells converted to PDF format. The filename should be `netid_assignment2_output.pdf`.
- A brief report in PDF format using this template[5], with name `netid_assignment2_report.pdf`.

[4] `https://www.kaggle.com/t/e0e164b02e9848e78473497580bd8ebf`

[5] `https://docs.google.com/document/d/10sJ1qHxXcIWdPkI-dvqLRNYYUE5wJIWEK4_jfMaHIxY/edit?usp=sharing`