Assignment Instructions:

In this assignment, you will implement Fully-Connected Neural Networks and Convolutional Neural Networks for image classification models. The goals of this assignment are as follows:

- Understand the layered architecture of Neural Networks and its components like activation functions, loss functions, and optimizers.
- Understand the architecture of Convolutional Neural Networks and get practice with training these models on data.

There are four parts to this assignment:

P0: Simple Classifiers

The notebook **simple_models.ipynb** will walk you through basic experiments (already coded) of linear classifiers and KNN classifiers. You are supposed to understand and analyze the results of those experiments. You will be asked to answer a few questions on the same. Note: You can modify the hyper-parameters to see if your results improve and make that a part of your analysis.

P1: Fully Connected Neural Network

The notebook **fully_connected_networks.ipynb** will walk you through implementing Fully-Connected Neural Networks. We have used a modular approach to code each essential element of the fully connected neural networks. You are also supposed to read the supplementary reading provided throughout the notebook. It will help you in the Q/A part of the assignment. You are supposed to complete all the TODOs in the **fully_connected_networks.py** file and submit the file and model weights to Gradecope.

P2: Convolutional Neural Network

The notebook **convolutional_networks.ipynb** will walk you through implementing Convolutional Neural Networks. Same as part 1, first you'll implement components of the convolution network which will be followed by implementing a three-layered network for classification. You are supposed to complete all the TODOs in the **convolutional_networks.py** file and submit the file and model weights to Gradecope.

P3: Fine-tuning Pretrained Network

The notebook **transfer_learning.ipynb** will demonstrate transfer learning using ResNet-18 on the CIFAR-10 dataset. You are supposed to upload the notebook to Colab, complete the TODOs in the notebook and upload only the trained model weight to Gradescope.

Steps:

1. Download the Assignment2 folder from Canvas.
2. Unzip the file. All files should appear in a folder named Assignment2.
3. Upload this ‘Assignment2’ folder on your Google Drive.
4. Double-click on the **simple_models.ipynb,** it should open this notebook in Google Colab. (If it doesn’t then click on ‘connect more apps’ and install Collaboratory). Go through this notebook.
5. Next, you open the **fully_connected_networks.ipynb** the same way. Follow the notebook line by line. The first step in this is to mount Google Drive. Once you do that you can access and edit your ‘.py’ files on Colab.
6. Open the **fully_connected_networks.py** file in the colab workspace side-by-side and proceed with the assignment
7. Once you are done with Part 1, follow steps 4 to 6 for the Part 3 as well.
8. Lastly, open the **transfer_learning.ipynb,** there is no ‘.py’ file for this section. Just complete the TODO in the notebook

Submission

You are supposed to submit in total 5 files for this assignment.

1. fully_connected_networks.py
2. convolutional_networks.py
3. fcn_model.pth
4. final_threeconvnet.pth
5. resnet.pth
6. Q/A report (PDF file containing your answers to the given below questions)

IMPORTANT NOTE: Do not change the names of either of the above files. For ‘.py’ files, put your code only inside the given TODO space.

Grading: (of Total 16)

Individual component grading schema within each part of the assignment would be present with the assignment. The final scores from each part will be scaled to the final scores as below.

P0: Ungraded (results might be helpful in Q/A)
P1: 5/16
P2: 6/16
P3: 2/16
Q/A: 3/16

Questions

You need to answer these questions in brief, using evidence from the output of your assignment, and submit it as a Q/A report. Please add your GTID and GaTech mail on top of the report and name the file “lastname_assignment2.pdf” (example: “burdell_assignment2.pdf”). PDF is the only accepted upload format for the report. The report should be strictly less than equal to 4 pages including any references you may want to provide. Feel free to use any template you like as long as the report is readable.

1. (1 point) Compare and Contrast the results of the Linear Classifier, KNN CLassifier, Fully Connected Classifier (your optimal solution), CNN Classifier (your optimal solution), and Transfer Learning (using ResNet-18). Why did you get the results you got (Write one line each about the classifier, as to why they performed well and didn’t perform well)

2. (1 Point) With respect to Part 1, briefly explain how different regularization parameters and different optimization techniques affect the results of your training and validation set. (Mention 2 points each on your findings with regularization and optimization techniques. You can use any relevant part of the assignment for context)

3. (1 Point) With respect to the final model that you submitted in Part 2, how many algebraic operations (additions and multiplications) were performed in the forward pass for the first layer (Conv_ReLU_Pool)? Explicitly mention input image dimensions, number of filters, and size of the filter along with your calculations. For simplicity assume only one image is passed.

4. (1 Point) For Part 2, justify your choice of hyperparameters based on the observations you made while playing with the model.

5. (1 Point) In Part 3, what changes do you observe when you continue training for longer? Comment on the training and validation losses and accuracies. What is this observed phenomenon called? What would your approach be to help with this?

6. (1 Point) You have learned about regularization in CNNs. However, not all networks use the same regularization parameters. Comment on the strength of regularization needed to train a model effectively, as the size of the model increases.