Fall 2021 CSC 412/512

Programming Assignment 3: Multi-Layer Neural Networks

Type:

Group Project (2~3 members). For group formation method, please see Attachment 1- PA3 Group Instruction. Indepedent work is only approved for extraordinary cases. Communication and leadership skills are expected and inherently examined in this project.

Due date:

Satuarday, **December 1st, 11:59:59PM**

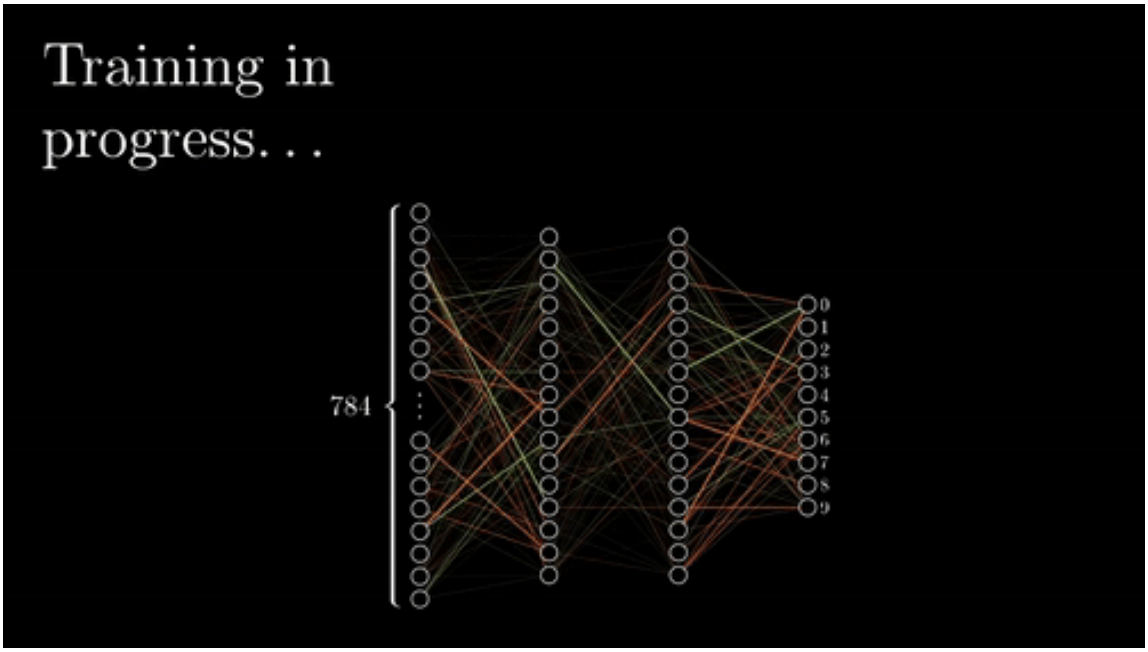
Group formation due date:

Wednesday, **November 10th, 11:59:59PM**

Grade:

100 marks in total, while it accounts for **25%** towards the final grade

Credit: UIUC Svetlana Lazebnik's CS 498 and Stanford's CS231n



Source: https://gfycat.com/gifs/detail/MiniatureDependentCob, https://www.youtube.com/watch?v=aircAruvnKk

In this assignment you will implement multi-layer neural networks on the CIFAR-10 image classification dataset. The goal of this assignment is to help you understand the fundamentals of neural networks and backpropagation. Specifically, you will write your own backward pass and

train two- and three-layer networks with ReLU and sigmoid nonlinearities. You will also get experience with hyperparameter tuning and using proper train/test/validation data splits. The format of this assignment is inspired by the Stanford CS231n assignments, and we have borrowed some of their data loading and instructions.

Download the starting code here.

The top-level notebook (`neural_network.ipynb`) will guide you through all the steps of training a neural network on CIFAR-10. You will implement the multi-layer neural network in the `neural_net.py` file.

We also provide you with a notebook to help with debugging and testing your neural network implementation by using a toy dataset along with numeric gradient checks. It is found in `develop_neural_network.ipynb`. **NOTE: This file is only for debugging/experimenting/testing purposes. It will not be graded, and you are not required to submit it.**

None of the parts of this assignment require use of a machine with a GPU. You may complete the assignment using your local machine or you may use Google Colaboratory. However, we encourage you to try using Colaboratory if you are not familiar with it and don't have access to GPUs since Colaboratory provides free access to a Tesla K80 (for running short jobs).

Environment Setup (Local)

If you will be completing the assignment on a local machine then you will need a python environment set up with the appropriate packages.

We suggest that you use Anaconda to manage python package dependencies (https://www.anaconda.com/download). This guide provides useful information on how to use Conda: https://conda.io/docs/user-guide/getting-started.html.

Data Setup (Local)

Once you have unzipped the zip file, navigate to the cifar10 directory in assignment3_materials and execute the get_dataset script provided:

```
cd assignment3_materials/cifar10/
./get_datasets.sh
```

Data Setup (For Colaboratory)

If you are using Google Colaboratory for this assignment you will need do some addition setup steps.

Download the assignment zip file and follow the steps above to download CIFAR-10 to your local machine. Next, you should make a folder in your Google Drive to hold all of your assignment files and upload the entire assignment folder (including the cifar10 dataset you downloaded) into this Google drive file.

You will now need to open the Assignment 3 ipython notebook file from your Google Drive folder in Colaboratory and run a few setup commands. However, we have condensed all the important commands you need to run into an **ipython notebook** (you can disregard the section on GPU setup for this assignment).

IPython

The assignment is given to you in the `neural_network.ipynb` file. As mentioned, 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:

You have the following four experiments to complete:

- 2-layer Relu
- 3-layer Relu
- 2-layer Sigmoid
- 3-layer Sigmoid

Submission of this assignment will involve the following steps:

1. You must upload three files to **Canvas**.
    1. All of your code (python files and ipynb file) **in a single ZIP file**. The filename should be **Student_IDs_PA3_code.zip**.
    2. Your ipython notebook with output cells converted to **PDF format**. The filename should be **Student_IDs_PA3_output.pdf**.
    3. A brief report in PDF format using this template. The filename should be **Student_IDs_PA3_report.pdf**.
    Each group's report needs to include a brief statement of individual contribution, i.e., which group member was responsible for which parts of the solution and submitted material. Don't forget to include the names of all group members at the top of the report.
    4. Please zip all the above documents into one package and name it as **Student_IDs_PA3.zip** and upload it to **Canvas**. Multiple attempts will be allowed but only your last submission will be graded. No email submission is accepted. We reserve the right to take off points for not following directions.

Multiple attempts will be allowed but only your last submission will be graded. No email submission is accepted. I reserve the right to take off points for not following directions.

Late Policy:

For every day that your assignment is late, your score gets multiplied by 0.75. The penalty gets saturated after four days, that is, you can still get up to about 32% of the original points by turning in the assignment at all. If you have a compelling reason for not being able to submit the assignment on time and would like to make a special arrangement, you must send me email at least

four days before the due date (any genuine emergency situations will be handled on an individual basis).

Please refer to our course syllabus on academic integrity, and extension requests.