Spring 2023 CS 747 Deep Learning
Assignment 1

Due date: February 14, 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 does **not** require any deep learning packages (e.g., tensorflow, pytorch), and it does not require GPU resources. You may use Google Colab or local machine to finish this assignment.

1 Multi-class Classification: Empirical Study (100 points)

In this problem, you will implement simple linear classifiers on the well-known CIFAR-10[1] image classification dataset. The goal of this assignment is to help you understand the fundamentals of a few classic methods and become familiar with scientific computing tools in python. You will also get experience in hyperparameter tuning and using proper train/test/validation data splits. The start code is provided, and you need to implement several python functions for multi-class classification, and write a report to summarize your observation and compare the performance of different methods.

You will implement the following classifiers (in their respective files):

- Perceptron (`Perceptron.py`)
- SVM (`SVM.py`)
- Softmax (`Softmax.py`)

The notebook (`CS747_DL_Assignment-1.ipynb`) will guide you through all the steps.

[1] `https://www.cs.toronto.edu/~kriz/cifar.html`

2 Support Vector Machine (bonus: 20 points)

- (10 points) For the SVM classifier in the Problem 1, we assume the training data is linearly separable. A dataset is linearly separable if any two classes in this dataset can be specified by finitely many linear classifiers. Is the CIFAR-10 data really linearly separable? Write a program to check the linear separability of CIFAR-10 dataset. (Hint: you can start by randomly choosing data from two classes and check if there exists one linear classifier which can correctly classify every data point.)
- (10 points) Construct a synthetic dataset with 2 classes which is *highly non-linearly separable*. Run your support vector machine in `svm.py` and report their best classification accuracy. If you are asked to improve the performance on this dataset, what would be your solution? Please design and implement a better algorithm than `svm.py` (it cannot be the same as `Perceptron.py` or `Softmax.py`).

Instructions about Environment Setup

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[2]. This guide provides useful information on how to use Conda[3].

Data Setup (Local) Once you have downloaded and opened the zip file, navigate to the cifar10 directory in assignment1 and execute the `get_datasets` script provided:

```
$ cd CS747-assignment1/cifar10
$ ./get_datasets.sh
```

or

```
$ cd CS747-assignment1/cifar10
$ python3 get_datasets.py
```

Data Setup (Google Colab) If you are using Google Colaboratory for this assignment, all of the Python packages you need will already be installed. The only thing you need to do is download the datasets and make them available to your account.

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

[2] `https://www.anaconda.com/download`

[3] `https://conda.io/docs/user-guide/getting-started.html`

and upload the entire assignment folder (including the datasets you downloaded) into this Google drive file.

You will now need to open the assignment 1 IPython notebook file from your Google Drive folder in Colaboratory and run a few setup commands. You can find a detailed tutorial on these steps here[4] (no need to worry about setting up GPU for now).

IPython The assignment is given to you in the `CS747_DL_Assignment-1.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[5]. You may then navigate to the assignment directory in the terminal and start a local IPython server using the jupyter notebook command.

Submission Instructions

- You must submit your output Kaggle CSV files from each model on the CIFAR-10 dataset to their corresponding Kaggle competition webpages:
  - Perceptron: `https://www.kaggle.com/t/666f427921c64a708919282846c7f4b7`
  - SVM: `https://www.kaggle.com/t/7d603b7e30e94fc7898b66f8093bbd90`
  - Softmax: `https://www.kaggle.com/t/96e7234d53db4747859025d39c157185`

  The baseline accuracies you should approximately reach are listed as benchmarks on each respective Kaggle leaderboard.
- You must upload three files on GMU Blackboard Portal:
  - 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_assignment1_code.zip`.
  - Your ipython notebook with output cells converted to PDF format. The filename should be `netid_assignment1_output.pdf`.
  - A brief report in PDF format using this template[6], with name `netid_assignment1_report.pdf`.

[4] https://medium.com/deep-learning-turkey/google-colab-free-gpu-tutorial-e113627b9f5d

[5] https://ipython.org/install.html

[6] https://docs.google.com/document/d/1J16WIJUE3QAL_uK2mKipHQWOnX3ICQVRj5K_H7h5SNs/edit?usp=sharing