EECS 498-007 / 598-005
Deep Learning for Computer Vision
Fall 2020

Assignment 1

In this assignment, you will first learn how to use PyTorch on Google Colab environment. You will then practice putting together a simple image classification pipeline, based on the k-Nearest Neighbor, and finally will learn how to use Autograder for evaluating what you implement. The goals of this assignment are as follows:

- Develop proficiency with PyTorch tensors
- Gain experience using notebooks on Google Colab
- Understand the basic Image Classification pipeline and the data-driven approach (train/predict stages)
- Understand the train/val/test splits and the use of validation data for hyperparameter tuning
- Implement and apply a k-Nearest Neighbor (kNN) classifier
- Learn how to test your implementation on Autograder

This assignment is due on **Friday, September 11, at 11:59 PM EDT**.

Q1: PyTorch 101 (60 points)

The notebook **pytorch101.ipynb** will walk you through the basics of working with tensors in PyTorch. You are required to write code on **pytorch101.py**.

Q2: k-Nearest Neighbor classifier (40 points)

The notebook **knn.ipynb** will walk you through implementing a kNN classifier. Your implementation will go to **knn.py**.

Steps

1. Download the zipped assignment file

- Click here to download the starter code

2. Unzip all and open the Colab file from the Drive

Once you unzip the downloaded content, please upload the folder to your Google Drive. Then, open each `*.ipynb` notebook file with Google Colab by right-clicking the *.ipynb file. No installation or setup is required! For more information on using Colab, please see our Colab tutorial.

3. Open your corresponding *.py from Google Colab and work on the assignment

Next, we recommend editing your `*.py` file on Google Colab, set the ipython notebook and the code side by side. Work through the notebook, executing cells and implementing the codes in the `*.py` file as indicated. You can save your work, both `*.ipynb` and `*.py,` in Google Drive (click "File" -> "Save") and resume later if you don't want to complete it all at once.

While working on the assignment, keep the following in mind:

- The notebook and the python file have clearly marked blocks where you are expected to write code. **Do not write or modify any code outside of these blocks**.
- **Do not add or delete cells from the notebook**. You may add new cells to perform scratch computations, but you should delete them before submitting your work.
- **Run all cells, and do not clear out the outputs, before submitting.** You will only get credit for code that has been run.

4. Evaluate your implementation on Autograder

Once you want to evaluate your implementation, please submit the `*.py` and `*.ipynb` files to Autograder for grading your implementations in the middle or after implementing everything. You can partially grade some of the files in the middle, but please make sure that this also reduces the daily submission quota. Please check our Autograder tutorial for details.

5. Download .zip file

Once you have completed a notebook, download the completed `uniqueid_umid_A1.zip` file, which is generated from your last cell of the `knn.ipynb` file.

Make sure your downloaded zip file includes your most up-to-date edits; the zip file should include *pytorch101.ipynb*, *knn.ipynb*, *pytorch101.py*, *knn.py* for this assignment.

6. Submit your python and ipython notebook files to Autograder

When you are done, please upload your work to Autograder (UMich enrolled students only). Your ipynb files *SHOULD include* all the outputs.

EECS 498-007 / 598-005: Deep Learning for Computer Vision

Justin Johnson
justincj@umich.edu

Website for UMich EECS course