CSCI-GA 2271: Assignment 1

Due: Thursday, October 16, 2025 at 6:00 PM.

Please note that marks will be deducted if your .ipynb cells are not run and outputs are not visible. Ensure that all cells are run before submission. For each part, please make a separate zip file and submit it in the corresponding Gradescope submission.

1 Part 1: Filtering and Convolutions [30 pts]

In this part of the assignment, you will explore image filtering and edge detection techniques using the provided Jupyter notebook. You will apply various padding techniques, implement convolution operations from scratch, and experiment with gradient-based edge detection using 1D filters. All tasks will be performed on the images provided (`zebra.png` and `cameraman.png`). Follow the instructions in the notebook to complete each step, visualize the results, and analyze the effects of different filters and parameters on the images. Use text cells in the Jupyter notebook to provide any required analysis or answers in 1-2 lines. While submitting, please ensure that outputs for all parts are displayed.

2 Part 2: Forward-mode and backward-mode AutoDiff: [20 pts]

Let $f(X) = a(b(c(X)))$, $a \mapsto \mathbb{R}^m, b \mapsto \mathbb{R}^q, c \mapsto \mathbb{R}^p, X \in \mathbb{R}^n$ such that $f : n \mapsto m$, then

$$\begin{aligned} \frac{\delta f}{\delta X} &= \frac{\delta a}{\delta b}\frac{\delta b}{\delta c}\frac{\delta c}{\delta X} \text{ or,} \\ f' &= a'b'c' \text{ where } f', a', b', c' \text{are jacobian matricies} \end{aligned} \tag{1}$$

f' can be calculated using the chain rule if we know the derivatives at each step of the computation graph.

Complete the **ValueFwd** and **ValueBwd** classes in the `auto_diff.ipynb` jupyter notebook to implement Forward mode and Backward mode Auto Differentiation.

3 Part 3: Convolutional Neural Networks [50 pts]

- Navigate to `convolutional_networks.ipynb` and read through the ipynb instructions on how to code up a CNN from scratch using tensors. You are only required to modify `convolutional_networks.py` in the specified lines and run the cells in the .ipynb file. Make sure all the helper files and folders are in the same directory to avoid hiccups.
- `fully_connected_networks.py` contains helper functions and is not expected to be modified.
- We recommend editing your *.py file on Google Colab, setting the ipython notebook and the code side by side.
- Work through the notebook, executing cells and writing code in *.py, 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.
- **Submit your `convolutional_networks.ipynb` and `convolutional_networks.py` as a zip file. Do not include other helper files or saved models in your submission**