Assignment 1

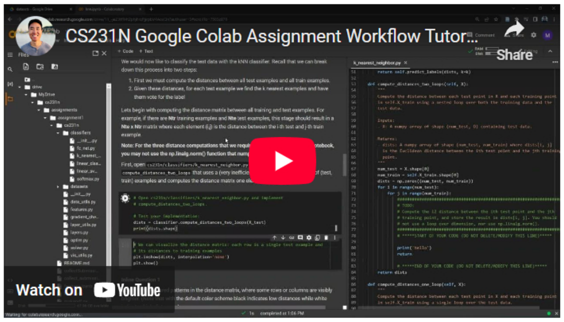
This assignment is due on **Wednesday, April 23 2025** at 11:59pm Pacific Time.

Starter code containing Colab notebooks can be downloaded here.

- Setup
- Goals
- Q1: k-Nearest Neighbor classifier
- Q2: Implement a Softmax classifier
- Q3: Two-Layer Neural Network
- Q4: Higher Level Representations: Image Features
- Q5: Training a fully connected network
- Submitting your work

Setup

Please familiarize yourself with the recommended workflow by watching the Colab walkthrough tutorial below:



Note. Ensure you are periodically saving your notebook (`File -> Save`) so that you don't lose your progress if you step away from the assignment and the Colab VM disconnects.

Once you have completed all Colab notebooks **except** `collect_submission.ipynb`, proceed to the submission instructions.

Goals

In this assignment you will practice putting together a simple image classification pipeline based on the k-Nearest Neighbor or the SVM/Softmax classifier. The goals of this assignment are as follows:

- 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**.
- Develop proficiency in writing efficient **vectorized** code with numpy.
- Implement and apply a k-Nearest Neighbor (**kNN**) classifier.
- Implement and apply a **Softmax** classifier.
- Implement and apply a **Two layer neural network** classifier.
- Implement and apply a **fully connected network** classifier.
- Understand the differences and tradeoffs between these classifiers.
- Get a basic understanding of performance improvements from using **higher-level representations** as opposed to raw pixels, e.g. color histograms, Histogram of Oriented Gradient (HOG) features, etc.

Q1: k-Nearest Neighbor classifier

The notebook **knn.ipynb** will walk you through implementing the kNN classifier.

Q2: Implement a Softmax classifier

The notebook **softmax.ipynb** will walk you through implementing the Softmax classifier.

Q3: Two-Layer Neural Network

The notebook **two_layer_net.ipynb** will walk you through the implementation of a two-layer neural network classifier.

Q4: Higher Level Representations: Image Features

The notebook **features.ipynb** will examine the improvements gained by using higher-level representations as opposed to using raw pixel values.

Q5: Training a fully connected network

The notebook **FullyConnectedNets.ipynb** will walk you through implementing the fully connected network

The notebook **FullyConnectedNets.ipynb** will walk you through implementing the fully connected network.

Submitting your work

Important. Please make sure that the submitted notebooks have been run and the cell outputs are visible.

Once you have completed all notebooks and filled out the necessary code, you need to follow the below instructions to submit your work:

1. Open `collect_submission.ipynb` in Colab and execute the notebook cells.

This notebook/script will:

- Generate a zip file of your code (`.py` and `.ipynb`) called `a1_code_submission.zip`.
- Convert all notebooks into a single PDF file.

If your submission for this step was successful, you should see the following display message:

```
### Done! Please submit a1_code_submission.zip and a1_inline_submission.pdf to Gradescope. ###
```

2. Submit the PDF and the zip file to Gradescope.

Remember to download `a1_code_submission.zip` and `a1_inline_submission.pdf` locally before submitting to Gradescope.