

# Deep Learning Assignment 1

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 Multiclass Support Vector Machine ( **SVM**) classifier
- implement and apply a **Softmax** classifier
- implement and apply a **Two layer neural network** classifier
- understand the differences and tradeoffs between these classifiers
- get a basic understanding of performance improvements from using **higher-level representations** than raw pixels (e.g. color histograms, Histogram of Gradient (HOG) features)

## Working on the assignment:

**Setup your Python environment:** You will need to set up a [virtual environment](#) for the project. To set up a virtual environment, run the following:

```
1 cd assignment1
2 sudo pip install virtualenv          # This may already be installed
3 virtualenv .env                     # Create a virtual environment
4 source .env/bin/activate            # Activate the virtual environment
5 pip install --upgrade pip
6 pip install -r requirements.txt      # Install dependencies
```

After you're done working on the project, run the following to deactivate the virtual environment:

```
1 deactivate
```

Don't forget to activate your virtual environment the next time you'll work on this project!

**Download data:** You will need to download the CIFAR-10 dataset. Run the following from the assignment1 directory:

```
1 cd cs231n/datasets
2 ./get_datasets.sh
```

**Start IPython:** After you have the CIFAR-10 data, you should start the IPython notebook server from the assignment1 directory. If you are unfamiliar with IPython, you should read this [IPython tutorial](#).

## Submitting your work:

Once you are done working run the `collectSubmission.sh` script; this will produce a file called `assignment1.zip`.

### Q1: k-Nearest Neighbor classifier (20 points)

The IPython Notebook `knn.ipynb` will walk you through implementing the kNN classifier.

### Q2: Training a Support Vector Machine (25 points)

The IPython Notebook `svm.ipynb` will walk you through implementing the SVM classifier.

### Q3: Implement a Softmax classifier (20 points)

The IPython Notebook `softmax.ipynb` will walk you through implementing the Softmax classifier.

### Q4: Two-Layer Neural Network (25 points)

The IPython Notebook `two_layer_net.ipynb` will walk you through the implementation of a two-layer neural network classifier.

### Q5: Higher Level Representations: Image Features (10 points)

The IPython Notebook **features.ipynb** will walk you through this exercise, in which you will examine the improvements gained by using higher-level representations as opposed to using raw pixel values.