DSE 220: Machine learning

Lab 4 — Deep learning

Many thanks to Dyuthi Sristi for creating this lab.

1. *A computer vision classification task.* In this problem, we'll train a classifier that takes as input an image of an ant or a bee, and determines which it is.

   Download `lab4-1.zip` from the course page and unzip it. You will see a Jupyter notebook, called `ants-bees.ipynb` and a directory called `hymenoptera_data`.

   (a) The `hymenoptera_data` directory has a subdirectory with the training set (`train`) and another with the test set (`val`). Take a look at some of the images. They are of varying sizes, with diverse backgrounds. And there are only 244 training images: not an easy learning problem!

   Run the first few cells of the notebook to see how the images are normalized in size. Pick an image from the directory and show both the original version and the normalized version.

   (b) Run the remaining cells of the notebook to see how the ResNet50 network is used to produce a 2048-dimensional representation for each image, and how a logistic regression classifier is constructed on top of this.

   Now, use this same 2048-d representation to construct a k-nearest neighbor classifier. Give the test accuracies obtained for $k = 1, 3, 5$.

2. *A two-dimensional classification task.*

   Download `lab4-2.zip` from the course page and unzip it. You will see a Jupyter notebook called `two-dim-classification.ipynb`. Run the cells one by one. The very last cell shows the decision boundary of the neural classifier at various points in the training process.

   (a) Plot how the training loss changes with epochs. Also, plot the training accuracy and validation accuracy in a single plot. Comment on the trends.

   (b) Effect of learning rate: run the code with `lr = [1, 0.1,0.01,0.001,0.0001]`. Plot the training and validation accuracy for each of these learning rates. Comment on which learning rate is best and what is the issue with other learning rates.

   (c) The current model uses 3 hidden layers with 20, 10, and 10 neurons in the first, second, and third hidden layers respectively. Now, keeping the remaining parameters the same, change this to a model with 1 hidden layer containing 100 neurons. Report the final training and testing accuracies for both the models. Also, report the number of network parameters for these models. Comment on which model is the best. Is the deep network better or the shallow one?