CNN Classifier Report
Yang Song

1, Starting with the network in cnn_mnist.ipynb, build a CNN with accuracy, precision, and recall all ≥ 99% on mnist_test.npy. I found that it took fewer than 20 epochs to arrive at such a CNN. Save your best model in an HDF5 file named cnn_mnist_99.hdf5.

I found choosing epoch=16 gave a pretty good solution. The test accuracy is 0.9913, and in the confusion matrix the precision, recall, and f1-score are all 0.991 > 99 percent.

2, In the baseline CNN, remove the dropout regularization layers and report what happens with the resulting model. I.e., is it better or worse than the baseline model in terms of accuracy, &c.? Explain what you see.

I removed the dropout regularization layers and tested on the case epoch=5. The accuracy increased up to more than 99 percent while the original one without removing the dropout layers had accuracy 98 percent.

3, Try some other modifications of the baseline CNN (e.g., adding more convolutional layers, changing the number of filters and each layer, image augmentation that we discussed in class) and report what happens. If you get a better model than the baseline model, save it as an HDF5 file named cnn_mnist_best.hdf5. In your written report, report what you tried and what you found.

I basically added more numbers to the layers and make them 64 and 128. It yields a better results than the previous one.

4, Build a "textbook" neural network for digit recognition using the example in cnn_mnist.ipynb as a model. Save this model in an HDF5 file named textbook_mnist.hdf5. In your written report, compare its performance to your CNNs. You may wish to increase the number of units in the hidden layer to improve performance.

I used the code provided in the textbook and saved the model. The performance is not as accurate as my previous CNNs, giving approximately 94 percent precision. I increased the conv2d number to 128, then the performance slightly increased.

5, One problem with training a CNN to recognize digits is that everything it sees it classifies as digit. In your written report, report what your classifier calls the images in nondigits.npy. (The four images in nondigits.npy are, in order, Hillary Clinton winking, Donald Trump, an American buffalo, and a Christmas tree.)

After running my code, I get the results 8, 3, 5, 0, for each picture.