CIS 472/572, Winter 2018
Homework 6: Neural Networks
DUE DATE: Submit via Google Collab by
Friday, March 23th at 11:59pm.

This assignment is **OPTIONAL**. If you complete it and submit it, it will take the place of your lowest homework score.

In this assignment, you will use Keras, a popular deep learning framework, to train several different neural networks on the MNIST digit recognition dataset and the CIFAR10 image recognition dataset.

1 Step 0: Google Collab

Go to Google Collab and create a new Python3 notebook to work in. You may want to switch to the GPU runtime by selected "Runtime," "Change runtime type," and "GPU."

You can access Google Collab here: `https://colab.research.google.com/`

2 Step 1: Complete a Keras Tutorial

This tutorial will walk you through installing Keras and using it to train a neural network for digit recognition on the MNIST dataset. Please complete the entire tutorial to learn the basics of Keras:

`https://elitedatascience.com/keras-tutorial-deep-learning-in-python`

You can use your Google Collab notebook for completing the tutorial.

3 Step 2: Train a Variety of Models

Next, use Keras to train several different models with the following architectures. All should have a softmax output layer with 10 nodes, corresponding to the 10 possible digit classes in MNIST. Use the commands from the tutorial as a starting point, but modify them as appropriate to create the following different models.

1. Multi-layer perceptrons with a single hidden layer:
   (a) `model1`: Multi-layer perceptron with one layer of 16 hidden sigmoid units (no dropout).
   (b) `model2`: Multi-layer perceptron with one layer of 128 hidden sigmoid units (no dropout).
   (c) `model3`: Multi-layer perceptron with one layer of 128 hidden rectified linear units (ReLU) (no dropout).

(d) `model4`: Multi-layer perceptron with one layer of 128 hidden ReLUs (50% dropout).

2. Convolutional neural networks:

   (a) `model5`: One 2D convolutional ReLU layer (as in the tutorial), one layer of 128 ReLUs (50% dropout).

   (b) `model6`: One 2D convolutional ReLU layer (as in the tutorial), one max pooling layer (as in the tutorial) (25% dropout), one layer of 128 ReLUs (50% dropout).

   (c) `model7`: Two 2D convolutional ReLU layers (as in the tutorial), one max pooling layer (as in the tutorial) (25% dropout), one layer of 128 ReLUs (50% dropout).

3. CIFAR10: Train two neural networks on the CIFAR10 image classification dataset. You can access the dataset with `from keras.datasets import cifar10`. You'll need to change the dimensions to 32x32x3 instead of 28x28x1, since the images are 32 pixels wide, 32 pixels high, and have 3 colors instead of 1.

   (a) `model8`: Multi-layer perceptron with one layer of 128 hidden ReLUs (50% dropout). (Similar to model4.)

   (b) `model9`: Two 2D convolutional ReLU layers (as in the tutorial), one max pooling layer (25% dropout), one layer of 128 ReLUs (50% dropout). (Similar to model7.)

After training your models, evaluate them on both the training and testing data. **The last cell in your notebook should print out the accuracy on the training and test data for each of the 9 models.**

4 Step 3: What To Submit

Click on "SHARE" and then enter two email addresses: `lowd@cs.uoregon.edu` and `nisansa@cs.uoregon.edu`.

Make sure that your notebook meets the specification described above – train 9 models, named `model1` through `model9`, and end the notebook by printing out the train and test accuracy for each one.

Have fun!