CMSC 25400/STAT 27725 — Homework 5

University of Chicago, Winter 2019 — due Tuesday, March 5 at 11:59pm

---

In this assignment you will code up a simple feed-forward neural network (NN) from scratch using only standard matrix/vector operators and train it by backpropagation to solve the full 10-digit MNIST handwritten digit classification task.

1. We recommend that you use the following architecture:
   (a) The input layer ($\ell = 0$) will consist of $785 = 28 \times 28 + 1$ neurons, of which the ouput of neuron $0 \leq i \leq 784$ is just the value of pixel i in the image, while the last neuron is a bias neuron whose output is always 1.
   (b) The hidden layers ($\ell = 1, 2, \ldots L-1$) will each have N_ℓ neurons.
   (c) The output layer ($\ell = L$) will have 10 neurons corresponding to the 10 classes.
2. Use a fully connected NN with the sigmoid transfer function, i.e. the vector of activations in layer $\ell = 1, 2, \ldots$ should be $a_\ell = \sigma(W_\ell \boldsymbol{a}_{\ell-1})$ where $\sigma(x) = (1 + e^{-x})^{-1}$.
3. For the outputs use the softmax loss $\ell(y, \boldsymbol{a}_L) = e^{[\boldsymbol{a}_L]_y} / (\sum_{i=1}^{10} e^{[\boldsymbol{a}_L]_i})$.
4. Train your neural network example-by-example with stochastic gradient descent, as discussed in class, or with minibatches of size b, with a fixed learning rate η. You will need to cycle through the training data multiple times (multiple epochs) and stop when error on the holdout set starts increasing substantially. The network should be started with random weights.
5. Training data is provided in the csv files `TrainDigitX.csv.gz` and `TrainDigitY.csv.gz` (if you use Python, the `loadtxt` function can automatically decompress these files). The test files are similarly named.
6. Experiment with varying the learning rate η, the minibarch size, the number of hidden layers, and the number of neurons in each layer, say $N_\ell \in \{32, 64, 128, 256\}$. For extra credit, you can also try coding up a convolutional neural net. For a comparison for how well different algorithms work for this data see `http://yann.lecun.com/exdb/mnist/`.
7. For this assignment you may use matrix libraries, but please do not use a neural network library or somebody else's implementation: the goal of the assigment is to give you the experience of coding up a neural network "from scratch".

Your writeup for this assignment should include the following:

1. A short description of your code and what choices you made during the implementation.
2. A study of how performance varies as a function of η, b, L, N_ℓ and the number of epochs. Try and optimize these parameters for the best performance on a hold-out set. Include plots for the error rate vs. each of these three parameters (with the other two set to reasonable values) on the test set.

In addition to your writeup please hand in the following:

1. Your full code in a form that the TAs can easily run it on the data if they want to verify it.
2. Your predictions on the two test sets `TestDigitX.csv` and `TestDigitX2.csv` (for the second one we do not publish reference labels).