Homework 9, due November 07th, 11:59pm

October 16, 2018

There are many good Neural Network packages that could be used for this project, including:

- MatConvNet - a Matlab version from U. Oxford (Windows/Linux/MacOS)
- Matlab - Matlab now has its own neural network library
- TensorFlow - Google's CNN package (Linux/MacOS)
- Theano - a Python version from U. Montreal (Linux/MacOS/Windows)
- CNTK - a C++ version from Microsoft (Windows/Linux)
- Caffe - a C++ version from Berkeley (Linux) with Python and Matlab interfaces

1. Using any of the packages mentioned above, any other package or your own implementation, perform the following tasks:

a) On the `miniboone` dataset, train a neural network with one hidden layer, with $k \in \{32, 64, 128, 256\}$ neurons in the hidden layer and ReLU activation functions for the hidden layer, and no activation function for the output layer. For each k find an appropriate learning rate and minibatch size to obtain a small final loss value on the training set after 100-300 epochs. Report in a table the misclassification errors for the four models on the training and test sets. Observe that since the `miniboone` data does not have a set training or test set, you should present results as the average of 10 independent random splits, each split using a random subsample of 80% of the data for training and the remaining 20% for testing. (3 points)

b) Repeat point a) with a neural network with two hidden layers, with 128 neurons in the first layer and $k \in \{32, 64, 128, 256\}$ neurons in the second layer and ReLU activation functions. (3 points)

c) Repeat point a) on the `madelon` dataset. For `madelon` you don't need to do the random splits, just use the training and test set from syllabus.(2 points)

d) Repeat point b) on the `madelon` dataset. (2 points)