CSCI 561: Foundations of Artificial Intelligence
Instructor: Prof. Laurent Itti
Homework #4: Artificial Neural Networks
Due on April 30 at 11:59pm, Los Angeles time, 2014

Problem Description

In this assignment, you will be creating an artificial neural network to identify images of handwritten digits.

Neural networks are a machine learning tool inspired by the brain. You will implement a simple neural network using fully-connected layers, and train using the backpropagation algorithm discussed in class. This means that for each image, you will do the following:

- Use the image to set the activations of your input layer
- Propagate the activations up to the top layer
- Compare the top layer activation with the image label
- Backpropagate the error (if any) down the network to adjust weights

Of course, for evaluation of new images, you will not perform the backpropagation stage.

As mentioned, the input will be images of handwritten digits (0-9). All images will be 28x28, with each pixel having a greyscale value between 0-255. The images are all stored in a single binary file. The labels will be a list of integers, representing which digit is being shown in the image. The labels are all stored in a separate binary file. Both of these files begin with a few metadata integer values (number of images, number of rows/columns, etc.). To help you concentrate on the neural network concepts, we are also providing Java code for reading these files into an easy-to-use Java class. This class also contains a method for visualizing the individual digit images. You may still use other languages. Translating the reading functions into C/C++/etc. should not be very difficult.

Your program's output is simply a list of the predicted values of each of the test images, as decided by your network. These values should all be saved in a single text file in the same order as they are in the test file. For example, if there are 5 test images and your program decides that they are images of the digits 1, 4, 9, 0, 3, then your output will be:

1
4

9
0
3

Your program will take 4 command-line arguments. These are the file paths to the training images, training labels, testing images, and output files respectively. Here's the breakdown of what your program must do:

- Construct/initialize a neural network
- Read the training data
- Use the training data to train the network (as described above)
- Read the testing data
- Evaluate the testing data using your network
- Output a text file containing the predictions for the testing data

Please ensure that your program does not take too long (~5 minutes) to perform all of these tasks.

We are providing separate image and label files for both training and testing. This will allow you to evaluate the performance of your network before submission.

We are not going to specify any of the parameters (number of layers, number of neurons per layer, learning rate, etc.) and instead will give you a target prediction accuracy of 85%. Full points will only be awarded if this accuracy is achieved on OUR testing set (this is not the same as the testing set you have been given). The goal of this is to allow you more freedom in this assignment! We encourage you to try out different combinations of parameters in order to improve your performance.

Grading:

The grader will compile your program using the following command, for examples:

C++: g++ *.cpp -o hw4

Java: javac *.java

And execute your program using, for examples for each experiment:

C++: ./hw4 train-images train-labels test-images output.txt

Java: java hw4 train-images train-labels test-images output.txt

Deliverables:

You are required to hand in all code that you wrote to complete this assignment; implementation language not important. However, if you code in C++ or Java, the TA

will be better able to assist you ☺. Also, we require that your code be able to run from the command line. Please include your source code, any additional files needed to compile it (makefiles, etc.), and a readme.txt to describe how to run your code, including the command(s) to run it.