CSE 5526 - Autumn 2014

Introduction to Neural Networks

Programming Assignment 1

Due Tuesday, Sep. 30

Grader: Yuzhou Liu
Email: liu.2376@osu.edu
Office: Dreese Labs 578
Office hours: Monday & Wednesday 3 - 4pm

Computer experiments play a critical role in the study of neural networks. Many aspects of this class can only be fully understood through computer simulations. Programming assignments help you gain first-hand experience with the algorithms introduced in the class. You may use any computer language for the implementation, but NOT a neural network toolbox.

Implement a two-layer MLP with backpropagation training. Train it to solve the parity problem: The network will be given an input vector of 1s and 0s and should output a single value: 1 if the input pattern contains an odd number of 1s and 0 if it contains an even number of 1s. Follow the implementation and training algorithm introduced in class and consult the textbook as necessary.

Use the following implementation details:

- Use 4 binary input elements, 4 hidden units, and one output unit
- Use a logistic sigmoid activation function ($\phi(v) = 1 / (1 + \exp(-v))$) for all units
- Initialize all weights and biases to uniform random numbers between -1 and 1
- Stop the learning procedure when an absolute error (difference) of 0.05 is achieved for every training pattern

Once you have implemented the model and training algorithm, perform the following experiments:

1. Vary the value of the learning rate η from 0.05 to 0.5 in increments of 0.05, and report the number of training epochs required to meet the stopping criterion for each choice of η.
2. Add a momentum term to the weight update with $\alpha = 0.9$ and report its effect on the speed of training for each value of η.
3. (Optional, extra credit) fix the learning rate and vary the number of input and hidden units. Report the number of training epochs required to meet the stopping criterion for each setting. Discuss patterns that emerge or "regimes" of behavior.

What you need to turn in:

1. A 1-2 page summary report, including 1-2 plots of relevant results
2. Your source code
3. A script or executable that can be run with no arguments to generate the plots in your report and any other evidence that your implementation is working properly