Math Modeling, Week 10

1. Hopfield networks and general (fully connected) Boltzmann machines can acquire *spurious attractors*, which are blends or averages of trained patterns. Given training patterns a^1, a^2, a^3, define $\bar{a}_i = \text{sign}(a_i^1 + a_i^2 + a_i^3)$. That is, $\bar{a}$ is determined by the majority among a^1, a^2, a^3.

Spurious attractors arise when training patterns are correlated across nodes. As a simple case, assume a^1, a^2, a^3 are generated according to some probability $p = \Pr[a_i^k = 1]$ for all nodes i and patterns k. To make things easy, you can assume these probabilities are perfectly reflected in the sample, meaning exactly $p^3 n$ of the nodes have $a_i^1 = a_i^2 = a_i^3 = 1$, exactly $p^2(1-p)n$ have $a_i^1 = a_i^2 = 1$ and $a_i^3 = -1$, etc.

(a) Define the network weights by Hebbian learning on all three patterns, and work out the energy for each a^k and for $\bar{a}$ as a function of p and the network size n. What values of p lead to $E(\bar{a}) < E(a^k)$? Work out the answer algebraically if you can, otherwise you can do it by simulation (the code should be simple enough: create patterns, define the weights, and calculate the energies).

(b) Think about psychological interpretations of this type of spurious attractor. What useful purpose might they serve? What would it mean for $E(\bar{a})$ to be less than $E(a^k)$?

2. Explore the code for the restricted Boltzmann machine on the MNIST dataset.

(a) Make sure you understand how the training routine (`RBMtrain.m`) works. See if you figure out the purpose of `momentum` (we'll discuss more in class).

(b) Use the script in `RBM_MNIST.m` to train the network on some number (`ntrain`) of patterns. Plot the hidden features (i.e., the map of connections from each hidden node to the visible nodes) and think about how interpretable they are.

(c) Test how well the network reconstructs the patterns it was trained on, and how well it generalizes to new (test) patterns. Is it better on training than test patterns, and if so why? How would you expect performance on training and test patterns to change as you vary the number of hidden units?

3. The MNIST dataset includes class labels (first column of both training and test files) that indicate the numeral (0-9) in each pattern. How could you use the RBM to predict these labels? Depending on how ambitious you feel, you can give a verbal description, formal equations, or working code.