University of Denver
Introduction to Artificial Intelligence
HW #4; Due midnight May 31, 2014

In this assignment you can choose one of two tasks:

1. Build code that performs linear regression (that is, a perceptron) using incremental weight updates. Given an input (vector) of **x**, the hypothesis we are learning is of the form ($x_0 = 1$):
$$h(\mathbf{x}) = \sum_{i=0}^{I} w_i \cdot x_i$$
Given a training example of (**x**, y), the update rule for the weights is:
$$w_i \leftarrow w_i + \alpha(y - h(\mathbf{x})) \cdot x_i$$
Attempt to learn the `XOR` of two variables using just the two variables as input. Then attempt to learning `XOR` using an additional input - the product of the two input variables. Write up the results and show a table for each case with the four test input points and the trained output in each case.

2. Build code that performs value iteration to learn the best policy in Figure 17.1 with various per-step rewards. (Assume 80% chance of moving in the intended direction and 10% chance of deviation at a right angle. Assume initially that each step has reward of -0.04.) For each state, maintain the utility of that state in an appropriate structure. (The terminal states have fixed utility.) Then, iterate through the state space updating the utility using:
$$U_{i+1}(s) \leftarrow R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s,a) U_i(s')$$
Draw the best policy for various per-step rewards and validate that your results are the same as the book. Try per-step rewards such as 0.0, -2.0, and -0.01.