CS 391L Machine Learning Final

This exam is open book. You may bring in your homework, class notes and textbooks to help you. You will have 1 hour and 15 minutes. Write all answers in the space provided. Please make sure YOUR NAME is on anything you turn in. Square brackets [] denote the points for a question.

1. **Graphical Models**

   (a) [15] Consider a tree-structured factor graph over discrete variables. Suppose we wish to evaluate the joint distribution $p(x_a, x_b)$ associated with two variables that *do not* have a common factor.
   Define a procedure for using the sum-product algorithm to compute this joint distribution. Hint: successively clamp one of the variables to each of its allowed values.

2. **Reinforcement Learning**

   In a standard reinforcement learning setting that can be described by a Markov Decision Process $MDP = \{S, A, T, R\}$ where S is the state space, A is the action space, T is the transition function and R is the reward function.

   (a) [5] *BRIEFLY* explain the difference between *Value Iteration* and *Q learning*.

   (b) [5] Suppose that you had an MDP defined for a two dimensional discrete state space $X \times Y$ and you doubled the resolution of the state space in each dimension. What would be the consequences for your MDP?

   (c) [15] To reduce the cost of high-dimensional search spaces one can approximate the value function with low degree-of-freedom parameterized functions. Suppose that the value function can be written as

   $$V(s) = \sum_{i=1}^{n} \theta_i b_i(s)$$

   where the functions $b_i(s)$ are known. The bootstrapping value function approximation method attempts to choose the θ_i so as to minimize the difference of samples

   $$(\hat{V}(s_{t-1}) - (r_t + \gamma \hat{V}(s_t))^2$$

   so how this can be done by computing the appropriate derivative.

3. **Genetic Algorithms**

   (a) [10] In a genetic algorithm a small set of examples or *fitness cases* are used to evaluate an individual. Why not just use all the possible inputs?

   (b) [5] Why is it useful to have fitness cases evolve?

   (c) [10] In genetic programming, could it ever be useful to constrain the crossover points? Give a reason why it might be a good idea and a reason it might not be.

4. **Games** In the classic game of chicken, two teenagers, Alice and Bob face off in opposing automobiles and drive madly towards each other. The idea is to get the other driver to swerve while not changing course (the 'straight' option). The payoff matrix is as follows.

		Alice	
		swerve	straight
Bob	swerve	0 / 0	1 / 0
	straight	0 / 1	-3 / -3

   (a) [10] A *Nash Equilibrium* is defined as a combined strategy where neither player can improve their own option by changing the move choice. Does the chicken game have any Nash equilibria?

   (b) [15] By analyzing the payoffs to individual players, derive the optimal probabilistic strategy.