CS 670: Programming Exercise 2

You are required to implement solutions to the Tic-Tac-Toe problem. One of the first issues to address is a suitable state-space representation.

1. For the first part of the assignment, you are required to design a policy gradient approach to solving this problem. Assume that in each of the states in the game, you have a preference value for each of the possible actions that you could take. Thus, for the initial position, you would have 9 possible preference values. These preference values are converted to probabilities by the use of the Boltzmann-Gibbs distribution.

   (a) Derive the update equations for the preference values, following the policy gradient approach. Recall the discussions in class: you may use the return from the start state as the quality of the policy.

   (b) Implement the policy gradient algorithm and train against different fixed opponents:

      i. One that picks the first free location, scanning row-wise from the top left

      ii. A random player

      iii. An optimal player

      Report learning curves giving the percentage of games **not lost** on the y-axis against the number of games played on the x-axis. Recall our discussion on learning curves in the class. Use 30 instances of the learning agent to generate these curves.

2. For the second part, you will explore value function based approaches.

   (a) Implement Q-learning with ϵ-greedy exploration and train against the same 3 fixed opponents used in the first part. Report learning curves averaged across 30 runs, as before.

   (b) Repeat above question with SARSA and ϵ-greedy exploration.

   (c) Use afterstates for the above problem and empirically compare the performance against the three fixed players. You may choose SARSA or Q-learning style updates. Explain the results.

You have to turn in the derivations asked above, the learning curves, well commented code, any other answers required[1], as well as a short description of your experience while conducting these experiments.

Additional Credit: Suggest an alternative parameterization of the policy and show theoretically or empirically the relative merits/demerits of the parameterization.

The due date for the homework is: **Friday, October 1st**.

[1] Such as a detailed description of the state representation, initialization of parameter values, etc.