CS747: Foundations of Intelligent and Learning Agents

Programming Assignment #2

Report

11 Oct, 2021

Deepanshu (190050032)

Department of Computer Science and Engineering
Indian Institute of Technology Bombay

2021-2022

Contents

Abstract

In this Assignment we try to solve the MDP problem using different techniques.
After that apply that to solve Anti-tic-tac-toe game and later find the optimal strategy for by watching our opponent.

1. Task1: MDP Planning Algorithm

1.1 Value iteration

Here we keep updating our value function using the Bellman Optimality Equation and we keep updating till value function can not be updated furthur significantly (theta = 1e-9).
And after that we found that optimal policy using this value function.

1.2 Haoward's Policy Iteration

Here we find the optimal value function just like we did in Value Iteration Algorithm. But on top of that we then update our policy and let it converge also. If it doesn't converge, we will repeat the same process till policy converge after convergence of value function.

1.3 Linear Programming

Here we used the Pulp module the algorithm but the basic idea was, we give it variables of our constraint equations and our objective function. And it will simply return the value solution.

2. Task2: Anti-Tic-Tac-Toe Game

2.1 encoder.py

Take State and Policy and convert it to MDP Problem

This file takes all the states of a player (player A) and strategy of our opponent (player B) and output the MDP file, which is basically we have converted the problem to a MDP problem.
The idea is to make move (all possible move from player A and check policy file for Player B) and check the situation of the game and find out the reward of each transition (with transition probability).

2.2 planner.py

Take MDP problem and give the optimal Valuefunction andd policy
.

We have already written this file. We will just run it over our new made MDP from encoder
Here the default algorithm will be run.

2.3 decoder.py

Take Valuefunction, policy and states, and give policy.

Here we have will check the policy and state and just format it in the way policy file should be

Now combining all three files, the abstraction we get is, give States and Policy file of our opponent and it will return us the optimal policy.

3. Task3: Anti-Tic-Toe Optimal Stratergy

This task is doing nothing from his side, instead it is calling the functions we implemented for task 2. But now we only have states and no policy file is there, we will generate a random one and try to find policy for player 1 and for player 2, till they converge.

PROOF Of CONVERGENCE :-
So here is the intution I got that if we have Player1-policy1 then we will train it and get player2-policy1 and then again train and get player1-policy2. Now since player2-policy came after training player1-policy1, and player1-policy2 came after player2-policy1 so the value function of player1-policy2 will be greater than or atleast equal to the player1-policy1. (Comparing policy here is same as we defined in the lecture).
Because if player1-policy2 came out to be worse we will instead try to keep player1-policy1, which is the equal to case.
And given the finite states and actions we can see that policy will indeed converge.