Markov Decision Processes

Yi Zhao

[Yzhao644@gatech.edu](mailto:Yzhao644@gatech.edu)

1. Introduction

This paper discusses methods to solve two Markov Decision Processes problems. Three methods, Value Iteration, Policy Iteration and Q-learning method were introduced to solve two MDP problems, a non-grid Forest Management Problem with larger states and a grid problem, Frozen Lakes with relative smaller number of states.

Further discussed are convergence analysis, hyper parameter’s individual effect on how the Q-learning can converge to optimal policy.

2. Problems

2.1 Frozen Lake

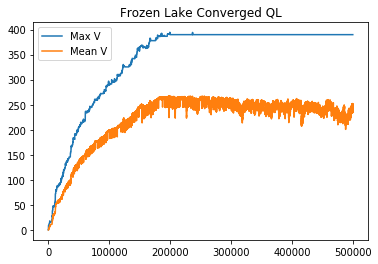
For the grid problem, I used Open AI Gym’s basic idea with some modification on the rewards. The grid is a square frozen lake that has several holes that agent can fall into. There is one grid which is the ultimate reward location. If the agent can reach that location, a final large reward is granted. When agent takes an action, there’s only 1/3 chance that the action will be taken, since the frozen lake is slippery, the agent might end up in the other directions. For up move, agent could move to left or right with 1/3 chance.

My

2.2 Forest Management

3. Value Iteration and Policy Iteration

4. Q-learning Algorithm

Chart, histogram

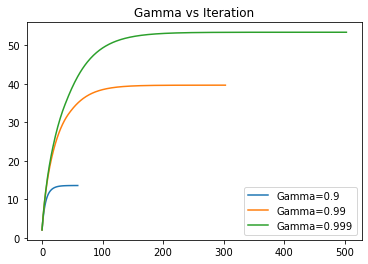
Description automatically generated

Chart, line chart

Description automatically generated

5. Solution for two problems

5.1 convergence

  
Diagram

Description automatically generated

delta convergence plot

5.2 Q-learning Hyper parameters: greedy method, epsilon greedy, all explore.

Forest management delta converge

A picture containing diagram

Description automatically generated

5.3

P3: Exploration Strategies

Plot error to see converge?

VI: Epsilon, Gamma

PI: Epsilon, Gamma

Q: gamma, alpha, alpha\_decay, alpha\_min, epsilon, epsilon\_min, epsilon\_decay, n\_iter

How does discount affect convergence?

I can see here higher discount takes more iterations to convergence ?

How is that so?

we should do this (vary the size of each MDP) because a grid world and non-grid world problem might scale differently in terms of how policy and value iteration handle it.

probs\_f, rewards\_f = hiive.mdptoolbox.example.forest(S=3000, p=0.001)

epsilon greedy

greedy

if ultimate reward is too small or gamma is too small and steps are long to get the final reward, they will choose to cut no matter what.