COMP0037 2022 / 2023 Robotic Systems

Lab 04: Monte Carlo Policy Prediction

COMP0037 Teaching Team

February 24, 2023

Overview

In this lab, we will look at implementing a couple of Monte Carlo algorithms for policy prediction. This is the empirical equivalent of policy evaluation. The evaluation is purely carried out through a series of trials. It is only based on material in Lecture 09.

The scenario is similiar to that in Lab 03: some designers have learned their lesson and bought a robot from the Slighlty Less McCheap Robot Company. The robot operates in an environment with goals and holes. Both are terminal sites — if the robot falls down them, it's game over. Goals give a big positive reward, holes a big negative one.

The robot is only capable of moving in four directions (`MOVE_LEFT`, `MOVE_RIGHT`, `MOVE_UP`, `MOVE_DOWN`) and waiting (`WAIT`). These first five actions are move actions. In addition, to these actions, we add two artificial ones. The `TERMINATE` action is only called at a terminal state (goal or hole) and `NONE` which is the only task allowed if the robot is in a state it shouldn't be able to get (e.g., inside a wall).

Installation Instructions

The code does not require you to install any additional packages beyond those required to support Lab 03.

-103	-102	-101	-100	-101	98	99	100	99	98

Figure 1: The value function computed by the policy evaluator.

Activities

1. The script `evaluate_simple_policy.py` is used to create a simple map and a simple policy. All the activities in this question ask you to use it. Both the robot and environment are deterministic.

   a. Run the script `evaluate_simple_policy.py` and confirm you get the value function shown at Figure 1. Can you guess what the underlying policy is which generates this state value function?

   b. The script `evaluate_simple_policy.py` uses an implementation of the on policy MC prediction algorithm with exploring starts to compute the state value function.

   By checking the implementation, why do you think it generates different values from the policy evaluation computed result? What do you need to do to fix it? Fix the code and check the results.

   Hint: Recall the definition of on policy and think about exploring starts.

   c. The script `evaluate_simple_policy.py` uses an implementation of the off policy MC prediction algorithm with exploring starts to compute the state value function. This currently crashes and does not wrong.

   What do you think is causing the difficulties? Have a go at fixing the code.

   Hint: Check the conditions which exist on the behaviour and target policies.