Reinforcement Learning - Assignment -Snakes and Ladders

Reinforcement Learning - Assignment - Snakes and Ladder

In this assignment, you will compare the performance of several reinforcement learning algorithms in a simple domain. You will also investigate the impact of different parameters on the performance of the reinforcement learning algorithm.

Important Dates

Assignment opened	25th November 2021
Deadline	**10th December 2021, 23:59**

Environment

This environment is based on a variation of the snakes and ladders game.

A sample game taken from ISLCollective is shown below. The player rolls a fair dice and then moves this many spots forward.

If a player reaches the bottom of a ladder, then s/he can move up to the end of the ladder. If a player reaches the head of a snake, s/he will slide down to the tail of the snake. For example, if you stop on 98, then you will be moved back to 79 (snake) and if you stop on 28, then you will be moved forward to 84 (ladder).

The game finishes if a player stops on the last spot. If a player rolls a throw that is too high, then they will bounce off the last spot. For example, if the last spot in a game is 100 and a player is currently at position 98 and rolls a 6, then they will move forward two positions (99, 100) and then bounce back another four positions (99, 98, 97, 96) and will finish their turn on position 96.

Create a game with 200 spots by combining two 100 spot boards and one with 300 spots by combining three boards with 100 spots.

Reinforcement Learning - Assignment -Snakes and Ladders

Snakes-and-Ladders-v0

Implement the game as an Open AI Gym environments so that you can apply a variety of reinforcement learning algorithms easily.

Use reinforcement learning to calculate the average number of steps that it takes to complete a snakes and ladder game.

Evaluate the performance of the standard TD(lambda) algorithm with lambda = 0,1, 0.5, and 0.9. Which one will learn the correct value function more quickly?

Snake-and-Ladders-v1

Then we extend the game to randomly generated dice, which are created randomly at the beginning of each batch of experiments. A die is created by selecting six random numbers from the set 1..6 with uniform probability.

Evaluate the performance of the standard TD(lambda) algorithm with lambda = 0,1, 0.5, and 0.9. Which one will learn the correct value function more quickly?

Determine suitable parameters of alpha and gamma. You will use the eps-Greedy search algorithm. Determine a suitable value for eps.

Compare the performance of standard value iteration and Q-Learning algorithms for this domain.

Snake-and-Ladders-v2

In this version of the game, we extend the Snake-and-Ladders-v1 game twofold.

Firstly, the player can choose to not take a ladder when they stop at the bottom of a ladder.

Secondly, a drone starts at location 50 and moves randomly one step forward or backward each round. If the drone reaches the end of the playing field Spot 1 or Spot 100, then it wraps around to Spot 100 or Spot 1 respectively. If a player lands on the spot with a drone, then the player will be picked up and moved randomly to a position +10/-20 away from the spot where the player was picked up.

Also, vary the number of dice that the agent can choose from two to six randomly created dice.

Evaluation

Evaluate the performance of value iteration and Q-learning with the following parameters

	Size of the Board		
	100 Spots	200 Spots	300 Spots
Snake-and-Ladders-v0	Lambda: 0.1, 0.5, 0,9	Lambda: 0.1, 0.5, 0,9	Lambda: 0.1, 0.5, 0,9
Snake-and-Ladders-v1 2 dice [1,1,3,3,5,5] and [2,2,4,4,6,6]	Lambda: 0.1, 0.5, 0,9	Lambda: 0.1, 0.5, 0,9	Lambda: 0.1, 0.5, 0,9

Ladders-v1 4 random dice	Lambda 0.1 0.5 0,9	Lambda 0.1 0.5 0,9	Lambda 0.1 0.5 0,9
Snake-and-Ladders-v1 6 random dice	Lambda 0.1 0.5 0,9	Lambda 0.1 0.5 0,9	Lambda 0.1 0.5 0,9
Snake-and-Ladders-v2 6 random dice, skip ladders, and drones	Lambda 0.1 0.5 0,9	Lambda 0.1 0.5 0,9	Lambda 0.1 0.5 0,9

Honesty Declaration

By submitting an assignment, you claim that your submission is **your own work only.** This means that you have developed the design, implemented the code, run the experiments, and all other work necessary to complete the assignment **by yourself**.

If you discussed your work with other students or used sources on the Internet, then you must say so clearly at the beginning of the assignment.

You can discuss parts of the assignment with others, but these discussions must be limited sharing ideas on a whiteboard or notepad. If you exchange code/programs/designs or exchange any material via the Internet, then you must say so in your honesty declaration.

Fill out the NTNU Honesty Declaration and add any external sources that you used during the assignment.

Submission

The submission consists of three parts: honesty, code, and report.

Step 1:
Before you can submit your assignment, you must send a message to the **@Online Lecturing Support** bot on the NTNU discord server (https://discord.gg/RuXRCGSXdC) via the command

```
!submit rl_a1 honesty
```

If you are registered for the course, then the submission should succeed and you will be able to proceed to the next step

Step 2:
Next create a directory with your id in the name for the code of your project.

Add a README.txt file into the directory, which shows: (a) how to compile and run your program, and (b) any interesting features and extensions of your assignment.

Create a zip file of your directory and send via a direct message to **@Online lecturing support** bot with the following command

```
!submit rl_a1 code
```

If your submission was successful, you should receive a response from the bot stating that the submission was successful.

Step 3:
Add a file <id>_report.pdf, which shows the results of the requested evaluations. Discuss the results. Were the results expected or did they surprise you? Did the results highlight shortcomings in the system? Do you have any ideas for how to fix those?

Submit your report via another message to the **@Online lecturing support** bot via the following command

```
!submit rl_a1 report
```

If your submission was successful, you should receive a response from the bot stating that the submission was successful.