# Reinforcement Learning (CSE 546 S23)

Assignment 1, Checkpoint 1 – A Rat’s life

Manudeep Herle

UBID: 50466613

UBIT name: Manudeep

1. Describe the deterministic and stochastic environments, which were defined (set of actions/states/rewards, main objective, etc).

#### Here is how I implemented my deterministic and stochastic environments:

For my environment – **A Rat’s life**, I’m passing an additional argument to the step function “prob” (0, 1] to control the stochasticity of the environment.

I generate a random number between 0 and 1, and whenever the random number is greater than the “prob” parameter, I generate another random number in the range of the possible action space (0 to 3 in this case). I assign this to my action variable to overwrite the intended action.

To run this environment in the stochastic mode, I change value of prob to .9. This way 10% of the times, the agent ends up in a different state than the one intended.

When the parameter “prob” is 1, the agent always moves to the same state for a given action from a given starting state.

Text

Description automatically generated

#### Actions

In both these environments I have 4 possible actions. Each episode has 12 timesteps.

1. Right
2. Left
3. Up
4. Down

#### States

There is a total of 16 states (4\*4) grid. The agent starts at (0,0) and must reach (3,3)

#### Rewards

There are totally 5 rewards in my grid world. Two of these are traps which impose a penalty of -5 on the agent.

Two others are cheese blocks which award a reward of +3 to the agent. Also, once the agent picks up the cheese, the reward disappears from the environment (that is cheese is no longer there for that episode).

And finally, there is the gold which gives the agent a reward of +10. The episode ends when the agent reaches the gold (3,3).

#### Main objective

The agent’s main objective is to get to the gold, at which point the episode ends. Gold is located at (3,3) and has a reward of +10. Agent would also try to maximize the reward along the way (collect as much cheese as possible) and avoid any traps.

1. Provide visualizations of your environments

Here are the images of my agent in the environment. Notice that some images are repeated, because the agent tried to go outside the grid and ended up not moving.

1. Logo

   Description automatically generated 2. Logo

   Description automatically generated

3Logo

Description automatically generated4. Logo

Description automatically generated

5.Logo

Description automatically generated 6. Logo

Description automatically generated

7. Logo

Description automatically generated 8. Logo

Description automatically generated

9. Logo

Description automatically generated 10. Logo

Description automatically generated with medium confidence

3. How did you define the stochastic environment?

For my environment, I’m passing an additional argument to the step function “prob” = (0, 1] to control the stochasticity of the environment.

I generate a random number between 0 and 1, and whenever the random number is greater than the “prob” parameter, I generate another random number in the range of the possible action space (0 to 3 in this case). I assign this to my action variable to overwrite the intended action.

To run this environment in the stochastic mode, I change value of prob to .9. This way 10% of the times, the agent ends up in a different state than the one intended.

When the parameter “prob” is 1, the agent always moves to the same state for a given action from a given starting state.

Text

Description automatically generated

4. What is the difference between the deterministic and stochastic environments??

Deterministic environments are those where the new state **s’** given an action **a** and a current state **s** is always the same. In stochastic environments the new state **s’** might not always be the same given an action **a** and current state **s**. i.e. in deterministic environments P(s’, r | s, a) = 1 or 0. In stochastic environments it can be anything in the range (0,1) (0 and 1 excluded).

5. Safety in AI: Write a brief review (∼ 5 sentences) explaining how you ensure the safety of your environments. E.g. how do you ensure that agent choose only actions that are allowed, that agent is navigating within defined state-space, etc.

I ensure my agent always acts in the defined action space by doing a few things. Firstly, the step function has 4 if statements defining what should be done if the agent takes any of the 4 allowed actions. There is no “else” statement and there is no other code in my script that can make the agent move.

Text

Description automatically generated

Secondly, whenever the agent’s action leads it to a cell outside the grid, I’m clipping it’s position (agent\_pos) so that the agent stays in the same position and doesn’t go outside the grid. Right now these two things ensure **100% guarante**e that the agent behaves as expected. When I do implement the tabular algorithm, I’ll be sure to add more checks.

Graphical user interface, text, application, chat or text message

Description automatically generated

Git expert:

Here’s are the 6 commits to my private repo on which ub-rl is a collaborator.

<https://github.com/manudeep96/RL-assignments>

Here is the github link to my repo.

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

References:

For the images, I used these sites. All the images are free for personal use (explicitly stated on the websites)

<https://lovepik.com/images/png-animal.html>

<https://www.cleanpng.com/png-mousetrap-trapping-clip-art-mouse-trap-646405/download-png.html>

<http://clipart-library.com/img/1734718.gif>

<https://pngtree.com/freepng/bright-gold-coin-decoration-material_4394118.html>

I referred to Dr. Alina’s base code, and also Nithin Kulkarni’s demo on matplotlib.

Additionally, I also referred to python, numpy and matplotlib documentation on their official sites.

<https://www.programiz.com/python-programming/break-continue>