CMSC 409: Artificial Intelligence

Fall 2023, Instructor: Dr. Milos Manic, http://www.people.vcu.edu/~mmanic

Project 3

CMSC 409: Artificial Intelligence
Project No. 3
Due Oct. 26, 2023, noon

Student certification:
Team member 1:
Print Name: _____ *Date:* _____
I have contributed by doing the following: _____
Signed: _____ *(you can sign/scan or use e-signature)*

Team member 2:
Print Name: _____ *Date:* _____
I have contributed by doing the following: _____
Signed: _____ *(you can sign/scan or use e-signature)*

Team member 3:
Print Name: _____ *Date:* _____
I have contributed by doing the following: _____
Signed: _____ *(you can sign/scan or use e-signature)*

Pr. 3.1 Understand and explore Reinforcement Learning (6pts)

1. Discuss what Reinforcement Learning (RL) is. (2pts)
2. What are the real-world applications which use RL, discuss 2 application areas. (2pts)
3. Discuss the advantages and disadvantages of RL. (2pts)

Pr. 3.2 Simulation of a self-driving cab using Q Learning (9 pts)

In this exercise you will learn:

1. How to run **Google Collab**:
2. The importance of reinforcement learning and how Q Learning can be used in solving a real world problem.
3. The importance of hyperparameters in reinforcement learning.

Problem description: In this exercise you will design a simulation of a self-driving cab using **Q Learning**. Consider an intelligent agent called "SmartCab". A SmartCab picks up the passengers at one location and drops them off at another. A SmartCab performs the following tasks:

1. Drops off the passenger to the right location.
2. Performs route optimization to minimize the ride time.
3. Observe traffic rules and prioritize passenger's safety.

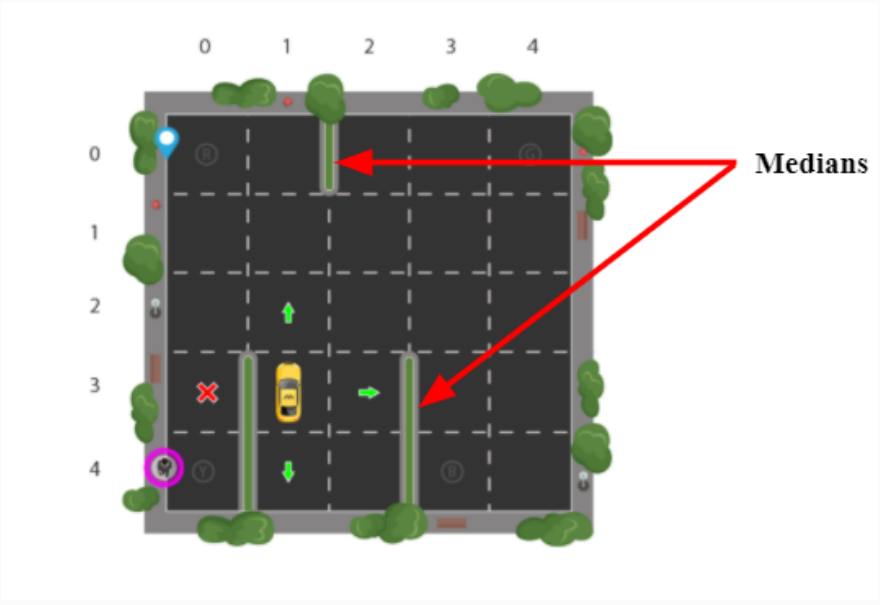
There are four locations labeled by letters. SmartCab picks up the passengers at one location and drops them off at another. The SmartCab receives +20 points for a successful drop-off and receives -1 point for every time-step it takes. There is also a -10 point penalty for illegal pick-up and drop-off actions.

CMSC 409: Artificial Intelligence

Fall 2023, Instructor: Dr. Milos Manic, http://www.people.vcu.edu/~mmanic

Project 3

The legal pick up and drop off locations are labeled as R, G, Y, B. (see the figure below). The SmartCab cannot drive through medians. Vehicle cannot go diagonally.



source: https://www.google.com/

There are different aspects to be considered while modeling an RL approach to this problem: 1. Rewards; 2. States, 3. Actions.

1. **Rewards**
   Since the SmartCab is reward-motivated and will learn by trial and error, here are some considerations to keep in mind when it comes to **rewards** and **penalties**:
   - The maximum number of time steps the agent can make is 8 (smaller city, see figure below).
   - The SmartCab should receive a high positive reward (+20) for a successful dropoff, because this behavior is highly desired.
   - The SmartCab should be penalized (-10) for illegal actions.
   - The SmartCab should be awarded a slight negative reward (-1) for each time step it takes to get to the destination. The smallest number of negative rewards deters SmartCab from taking routes longer than necessary (yes, this award is not ideally defined).

2. **State Space**
   In RL, the agent encounters a state, and then takes action according to the state it is in. The **State Space** is the set of all possible situations (states) our SmartCab could be in. The state should contain useful information the agent needs to make the best next action.

3. **Action Space**
   The allowable actions are: 1) move in a certain direction; 2) pick up a passenger, 3) drop off a passenger.

CMSC 409: Artificial Intelligence

Fall 2023, Instructor: Dr. Milos Manic, http://www.people.vcu.edu/~mmanic

Project 3

Do the following:

1. Change the following hyperparameters of the RL pipeline. Make a note of the reward for each combination of the hyperparameters. Discuss how the values of the hyperparameters affect the total reward for the smart cab **(6pts)**
    a. alpha: 0.01, 0.1, 0.9
    b. gamma: 0.40, 0.99
    c. epsilon: 0.5, 1

    Example: You can try:
    - alpha: 0.01
    - gamma: 0.40
    - epsilon: 0.5

    (then change alpha for the same gamma and epsilon, etc.)

2. Discuss the importance of reducing the epsilon parameter in achieving a higher reward **(1pts)**

3. Run the "*Smart_cab_brute_force.ipynb*" and compare and discuss the results with the reinforcement learning results. **(2pts)**

Extra Credit: (3pts)
Implement the *epsilon_decay* as:
*epsilon_decay = min_epilson +(max_epilson - min_epilson) * exp (-epsilon_decay_rate * episode)*

using the following hyperparameters:

- min_epilson
- max_epilson
- epsilon_decay_rate

1. Which values of these hyperparameters (alpha, gamma, epsilon, min_epilson, max_epilson, epsilon_decay_rate) result in the highest reward?
2. What is the highest reward?

Note:

1. Project deliverable should be a zip file containing:
    a. Written report (pdf) with answers to all of the questions above.
    b. Updated Source code in Ipython notebook file (.ipynb)
2. Submit your zip file to Canvas. Please name the zip file as GroupName_Project3.zip.