



Massachusetts Institute of Technology
Dept. of Electrical Engineering and Computer Science
Fmall Semester, 2018
[MIT 6.S198: Deep Learning Practicum](#)

Assignment 6: Reinforcement Learning with OpenAI Gym

<WRITEUP REQUIRED - Problem 1>

What is the reward scheme for FrozenLake?

You receive a 1 if you reach the goal, zero otherwise

<WRITEUP REQUIRED - Problem 2>

How big is the table used for Q-learning above?

The space is 16 by 4 for the 16 states and 4 actions.

Make a new file called **main.py** with the following code. Keep in mind that Google Docs handles quotes and indentation weirdly, so if you copy and paste you will need to fix those errors.

```
import gym
import numpy as np

env = gym.make("FrozenLake-v0")
po
#Q learning code starts here
Q = np.zeros([env.observation_space.n, env.action_space.n])

#set learning parameters, you will play around with these
lr = .8      #learning rate, how fast the Q-learning will learn at
gamma = .95  #discount factor, how much future rewards matter
num_episodes = 2000

#create list to contain total rewards and steps per episode
rewardsList = []
for i in range(num_episodes):
```

```

s = env.reset()          #reset environment and get first new observation
totalRewards = 0         #reward during this game
d = False                #is game finished
j = 0                    #index for checking number of steps you allow algo to run

#Q learning algo
while j < 99:            #let the learning happen after 99 steps per game
    j += 1
    #choose action by greedily picking from Q table
    a = np.argmax(Q[s,:] + np.random.randn(1,env.action_space.n)*(1./(i+1)))

    #Get new state and reward from environment
    new_state, reward, d, _ = env.step(a)
    #Update Q-Table with new knowledge
    Q[s,a] = Q[s,a] + lr*(reward + y*np.max(Q[new_state,:]) - Q[s,a])
    totalRewards += reward
    s = new_state
    if d == True:
        break
    rewardsList.append(totalRewards)

print(rewardsList)

```

<WRITEUP REQUIRED - Problem 3>

Did the rewards improve over time? Why or why not?

Yes as the game learns it starts with zero rewards and then gets 1s over time though not consistently because of the random noise and the number of moves it takes to get to 1.

<WRITEUP REQUIRED - Problem 4>

In the above code segment, we provided some values for the learning rate and discount factor. Try different values for the learning rate. What do you observe?

Discount <.5 results in all zeros

Discount <.75 mostly zeros

Discount ~.9 we get decent results

Learning rate <.5 mostly zeros

Learning rate ~.75 pretty good

Turn in this assignment here: <https://goo.gl/forms/0i0ZDn CZ6vmaapIE3>

After you're done with this assignment, feel free to try different environments that OpenAI Gym Provides! It's a great resource for your projects if you are using reinforcement learning.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).