

[강화 학습 특론]

Assignment 1: Model-Free Reinforcement Learning

1. 목표

- Open AI Gym frozenLake 환경에서 Model-Free RL 방법으로 Optimal Policy 찾기

2. 개발 환경 : 아래 평가 환경에서 동작 가능해야 함

- Python (version 3.6+) on Linux (ubuntu 18.04 LTS)

3. 템플릿 파일 및 제출 파일

- 제공되어지는 템플릿 파일 1개
 - frozen_lake.py: FrozenLake 환경 설정 및 기초 템플릿 코드 (제출하지 않음)
 - 템플릿 파일을 실행하면 agent.py 를 import 하는데 없어서 에러가 발생합니다. 해당 agent.py 을 구현하여 제출해야 합니다.

4. Open AI Gym FrozenLake 환경 소개

<https://gym.openai.com/envs/FrozenLake-v0/>

5. frozen_lake.py 설명 (수정하지 않으며 제출하지 않습니다.)

해당 파일을 실행하면 그림 1 과 같이 미끄러짐 유무와 맵 크기를 묻습니다.

가장 쉬운 난이도는 is_slippery = no , map_size = 4x4 입니다.

메뉴 1 번은 환경을 실행해서 키보드 0, 1, 2, 3 으로 상하좌우로 움직여 볼 수 있습니다.

2 번은 Q-learning 학습을 위해 Agent 클래스를 생성하고 학습을 요청 합니다.

3 번은 학습이 종료 된 Q 함수를 사용하여 테스트 합니다.

```

is_slippery no or yes : no
map_size 4x4 or 8x8 : 4x4

1. Checking Frozen_Lake
2. Q-learning
3. Testing after learning
4. Exit
select:

```

그림 1. 실행 옵션 설정

(참고: 가장 쉬운 난이도는 is_slippery = no, map_size = 4x4 입니다.)

6. agent.py 개발 설명

*** agent.py 파일을 만들고 q-learning 을 구현하고, 제출 시에는 파일명을 학번.py 로 변경해서 제출합니다.**

frozen_lake.py 코드를 먼저 확인하고 연동될 수 있도록 설계 및 구현하세요.

q-learning 알고리즘 구현을 포함하여 학습을 위한 episode_횟수, epsilon, alpha, gamma 와 같은 hyper parameter 값을 환경 설정에 맞게 잘 설정하는 것이 중요합니다.

q-learning 학습이 진행되고 있음을 확인할 수 있게 agent.learn() 함수 안에서 아래 로그를 episode 100번째마다 출력하세요.

```
print("\rEpisode {}/{} || average reward {}".format(cur_episode, max_episodes, avg_reward), end="")
```

cur_episode : 현재 에피소드 index

max_episode : 학습할 총 episode 개수

avg_reward : 최근 100 번 에피소드의 평균 reward

7. 채점 기준

- Total 100 points
 - 40 points: is_slippery=no, map_size=4x4 학습 후 메뉴 3 으로 테스트 결과가 1.0
 - 30 points: is_slippery=yes, map_size=4x4 학습 후 메뉴 3 으로 테스트 결과가 0.7 이상
 - 10 points: is_slippery=no, map_size=8x8 학습 후 메뉴 3 으로 테스트 결과가 1.0
 - 10 points: is_slippery=yes, map_size=8x8 학습 후 메뉴 3 으로 테스트 결과가 0.7 이상
 - 10 points: 보고서 - 개발환경, 구현한 메뉴/기능 실행 결과 명시.
조교는 보고서에서 명시된 항목에 대해서만 평가 합니다.

8. 제출 관련

- 제출 마감일: 3.21 (일) 23:59.
 - 지연 제출은 24시간마다 -15 점, 72시간 초과는 받지 않습니다.
 - 표절은 양쪽 모두 0점 처리 되며, 두 번 이상 반복 시에는 F 입니다.
- 구현한 agent.py 는 파일명을 “학번.py” 으로 제출하세요.
- 보고서 파일은 PDF 포맷으로 “학번.pdf”로 제출하세요.
- 두 파일을 iCampus 과제란에 각각 업로드 합니다.