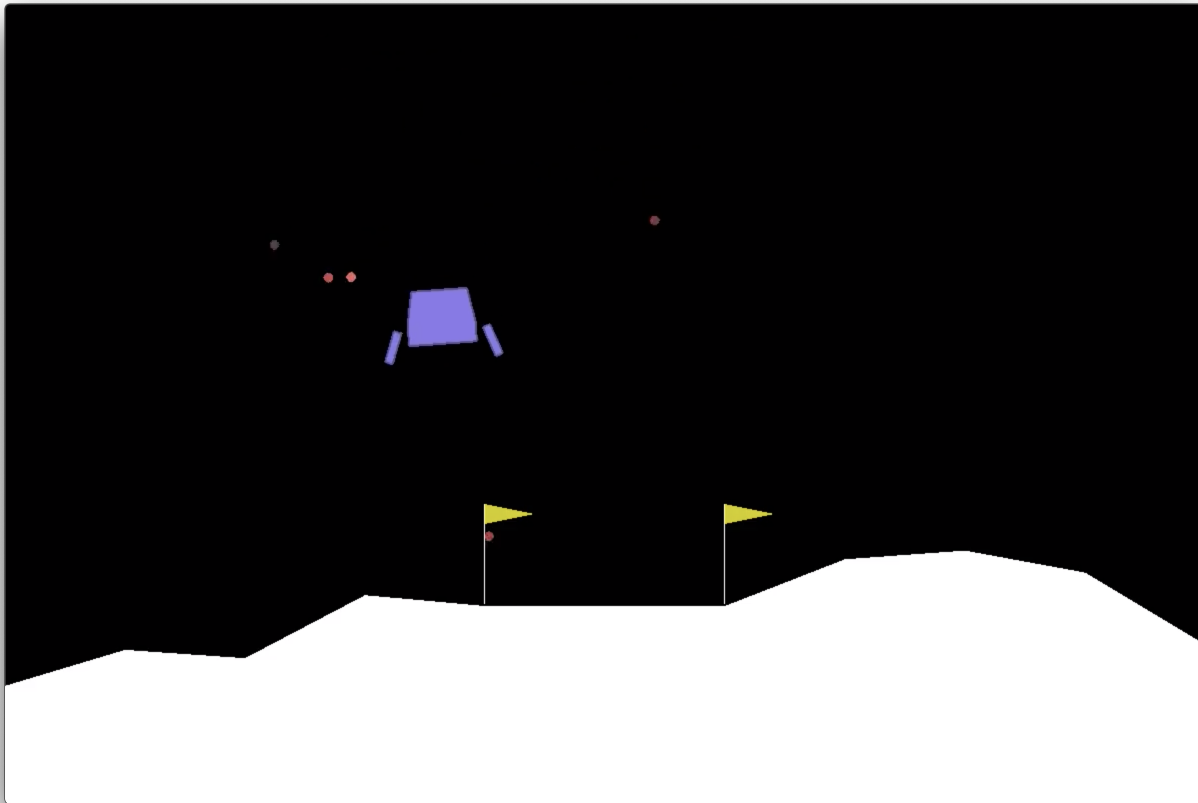


OpenAI Gym

LunarLander-v2 풀어보기

LunarLander-v2



LunarLander-v2

- observation은 8개 (continuous)
 - x pos, y pos
 - x-velocity, y-velocity
 - lander-angle
 - angular-velocity
 - right-leg grounded, left-leg grounded
- action은 4개 (discrete)
 - 아무것도 안하기, 왼쪽 회전 엔진 가동, 메인 엔진 가동, 오른쪽 회전 엔진 가동.

Reward

- (0,0)에 도착해야함.
- 에피소드는 우주선이 부딪치면 -100 리워드 더 받으면서 끝난다.
- 혹은 멈추면 +100 더 받으면서 끝난다.
- 각각의 다리가 랜딩패드에 닿으면 +10.
- 랜딩패드에 닿았어도 다시 떨어지면 리워드를 잃는다.
- 엔진을 켜는 것은 프레임마다 -0.3 reward
- 리워드 200 이상 받으면 solved

Box2D

- Box2D 환경은 python 3.5 버전을 사용하기 때문에 환경을 다시 만들었습니다.
- 메인함수에서 휴리스틱으로 푸는 모습을 확인할 수 있습니다.
- (휴리스틱으로 풀어도 대부분 reward 200 넘기며 성공)

```
# To see heuristic landing, run:  
#  
# python gym/envs/box2d/lunar_lander.py  
#  
# To play yourself, run:  
#  
# python examples/agents/keyboard_agent.py LunarLander-v2
```

Cross entropy method

- 시간이 매우 오래걸림...
- 마운틴카랑 다르게 에피소드당 시간제한이 없어서 뒤쪽으로 갈수록(훈련이 잘 될수록 죽지않기 때문에) 배치 하나 돌리는 시간이 오래걸림
- Google Colab에서 한번에 12시간씩 GPU를 무료로 사용할 수 있길래 이용하였습니다.
- $105 * 100$ 번의 에피소드로 solve.

Cross entropy result

```
71: loss = 0.932, reward_mean=77.9, reward_bound=129.0
72: loss = 0.925, reward_mean=57.7, reward_bound=133.0
73: loss = 0.931, reward_mean=64.4, reward_bound=141.8
74: loss = 0.930, reward_mean=68.3, reward_bound=126.4
75: loss = 0.916, reward_mean=74.8, reward_bound=140.1
76: loss = 0.909, reward_mean=81.9, reward_bound=148.2
77: loss = 0.906, reward_mean=67.2, reward_bound=142.9
78: loss = 0.879, reward_mean=62.4, reward_bound=133.9
79: loss = 0.884, reward_mean=58.3, reward_bound=138.7
80: loss = 0.845, reward_mean=73.2, reward_bound=149.4
81: loss = 0.829, reward_mean=85.9, reward_bound=164.0
82: loss = 0.786, reward_mean=94.6, reward_bound=187.1
83: loss = 0.775, reward_mean=85.7, reward_bound=193.4
84: loss = 0.758, reward_mean=130.1, reward_bound=238.1
85: loss = 0.741, reward_mean=123.5, reward_bound=241.1
86: loss = 0.759, reward_mean=131.0, reward_bound=254.6
87: loss = 0.743, reward_mean=144.3, reward_bound=255.5
88: loss = 0.754, reward_mean=135.9, reward_bound=251.9
89: loss = 0.707, reward_mean=131.2, reward_bound=251.8
90: loss = 0.720, reward_mean=98.9, reward_bound=209.2
91: loss = 0.748, reward_mean=117.8, reward_bound=245.3
92: loss = 0.731, reward_mean=139.2, reward_bound=261.9
93: loss = 0.747, reward_mean=176.8, reward_bound=264.9
94: loss = 0.757, reward_mean=153.8, reward_bound=259.2
95: loss = 0.750, reward_mean=160.7, reward_bound=257.8
96: loss = 0.751, reward_mean=183.1, reward_bound=266.6
97: loss = 0.732, reward_mean=200.0, reward_bound=263.3
98: loss = 0.748, reward_mean=182.8, reward_bound=267.5
99: loss = 0.773, reward_mean=179.9, reward_bound=264.8
100: loss = 0.773, reward_mean=164.1, reward_bound=265.3
101: loss = 0.734, reward_mean=188.5, reward_bound=267.4
102: loss = 0.742, reward_mean=194.5, reward_bound=265.8
103: loss = 0.712, reward_mean=199.6, reward_bound=271.8
104: loss = 0.714, reward_mean=194.6, reward_bound=268.0
105: loss = 0.729, reward_mean=216.4, reward_bound=268.5
Solved!
```

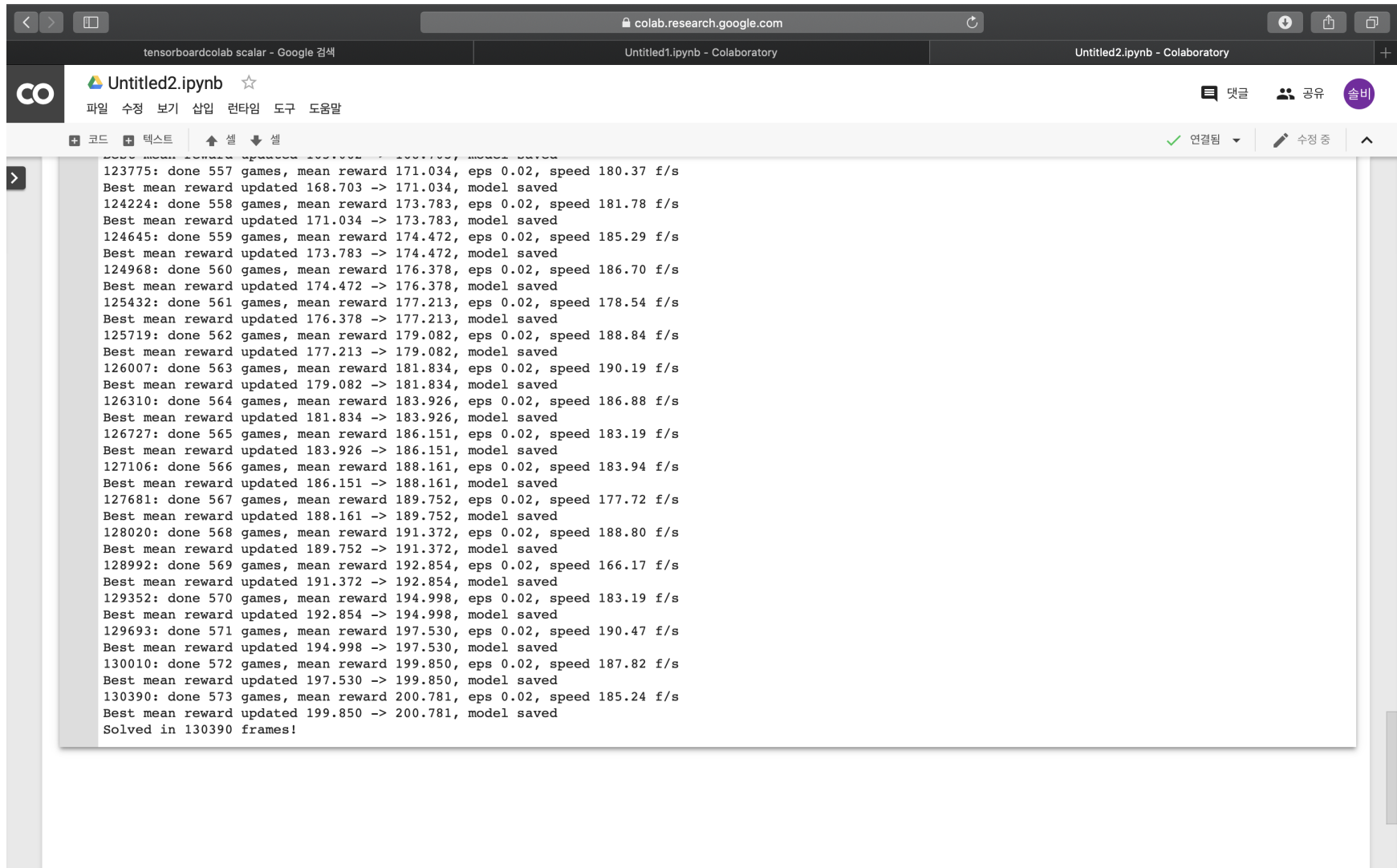
Value-iteration method & Tabular Q-learning

- table 방식에 적용하기에 LunarLander의 observation이 적절하지 않음 (continuous한 8개 value)
- Discretization wrapper를 이용하려고 했으나 observation value의 범위가 $-\text{INF} \sim \text{INF}$ 여서 새로 wrapper 를 만들어야 했음.
- observation 구간을 잘 나누어 유한 개의 state로 표현하더라도 $step^6 * 2 * 2$ 만큼의 state개수가 나오게 되어 연산이 오래걸림.

Deep Q-Network

- Mountain car 때와 동일한 네트워크 사용
- 128개 뉴런 사용한 2개의 hidden layer.
- Mountain car 를 풀때에는 cross entropy와 DQN 방식에서 풀기까지 걸린 시간이 비슷했는데, LunarLander 에서는 DQN 이 상대적으로 훨씬 적은 시간이 걸림.
- Cross entropy는 배치수만큼의 에피소드가 실행되어야 학습할 수 있는 반면(elite episode), DQN 은 step 단위로 학습하기 때문인 것으로 추정.
- LunarLander 처럼 에피소드 하나의 길이가 길고, 중간지표가 될 수 있는 적절한 reward가 있다면 cross entropy 보다는 DQN이 더 적합한 풀이 방식으로 보임.

DQN result & 학습 영상



The screenshot shows a Google Colab notebook titled "Untitled2.ipynb". The interface includes a top bar with navigation icons and tabs for "tensorboardcolab scalar - Google 검색", "Untitled1.ipynb - Colaboratory", and "Untitled2.ipynb - Colaboratory". The notebook content displays a series of training logs for a DQN model. Each log entry consists of a line number, a status (e.g., "done"), the number of games played, the mean reward, the epsilon value, the speed in frames per second (f/s), and a message indicating when the model was saved. The training progresses from 557 games to 573 games, with the mean reward increasing from 171.034 to 200.781. The final line indicates that the training was solved in 130390 frames.

```
123775: done 557 games, mean reward 171.034, eps 0.02, speed 180.37 f/s
Best mean reward updated 168.703 -> 171.034, model saved
124224: done 558 games, mean reward 173.783, eps 0.02, speed 181.78 f/s
Best mean reward updated 171.034 -> 173.783, model saved
124645: done 559 games, mean reward 174.472, eps 0.02, speed 185.29 f/s
Best mean reward updated 173.783 -> 174.472, model saved
124968: done 560 games, mean reward 176.378, eps 0.02, speed 186.70 f/s
Best mean reward updated 174.472 -> 176.378, model saved
125432: done 561 games, mean reward 177.213, eps 0.02, speed 178.54 f/s
Best mean reward updated 176.378 -> 177.213, model saved
125719: done 562 games, mean reward 179.082, eps 0.02, speed 188.84 f/s
Best mean reward updated 177.213 -> 179.082, model saved
126007: done 563 games, mean reward 181.834, eps 0.02, speed 190.19 f/s
Best mean reward updated 179.082 -> 181.834, model saved
126310: done 564 games, mean reward 183.926, eps 0.02, speed 186.88 f/s
Best mean reward updated 181.834 -> 183.926, model saved
126727: done 565 games, mean reward 186.151, eps 0.02, speed 183.19 f/s
Best mean reward updated 183.926 -> 186.151, model saved
127106: done 566 games, mean reward 188.161, eps 0.02, speed 183.94 f/s
Best mean reward updated 186.151 -> 188.161, model saved
127681: done 567 games, mean reward 189.752, eps 0.02, speed 177.72 f/s
Best mean reward updated 188.161 -> 189.752, model saved
128020: done 568 games, mean reward 191.372, eps 0.02, speed 188.80 f/s
Best mean reward updated 189.752 -> 191.372, model saved
128992: done 569 games, mean reward 192.854, eps 0.02, speed 166.17 f/s
Best mean reward updated 191.372 -> 192.854, model saved
129352: done 570 games, mean reward 194.998, eps 0.02, speed 183.19 f/s
Best mean reward updated 192.854 -> 194.998, model saved
129693: done 571 games, mean reward 197.530, eps 0.02, speed 190.47 f/s
Best mean reward updated 194.998 -> 197.530, model saved
130010: done 572 games, mean reward 199.850, eps 0.02, speed 187.82 f/s
Best mean reward updated 197.530 -> 199.850, model saved
130390: done 573 games, mean reward 200.781, eps 0.02, speed 185.24 f/s
Best mean reward updated 199.850 -> 200.781, model saved
Solved in 130390 frames!
```