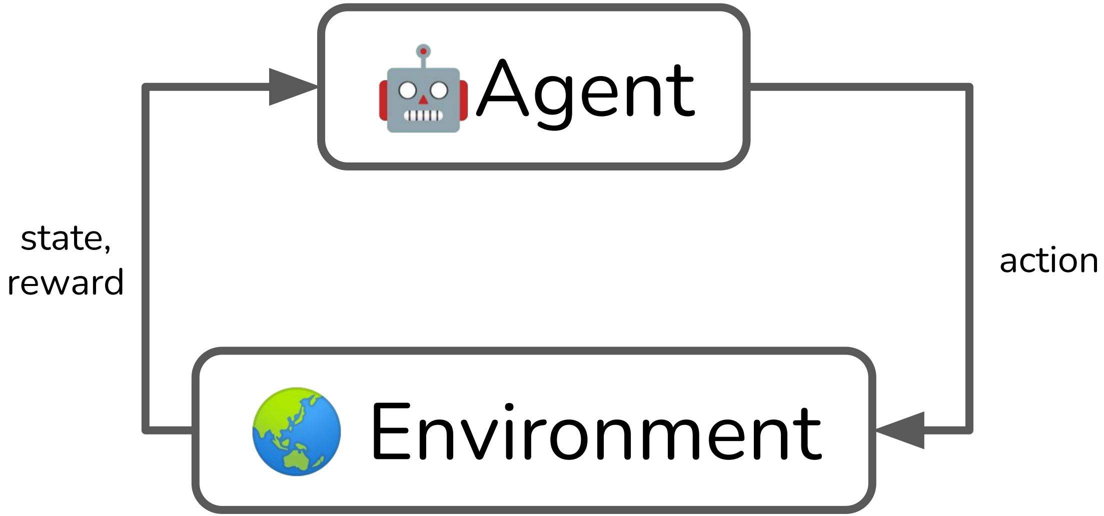
Learning Reinforcement Learning with OpenAI Gym

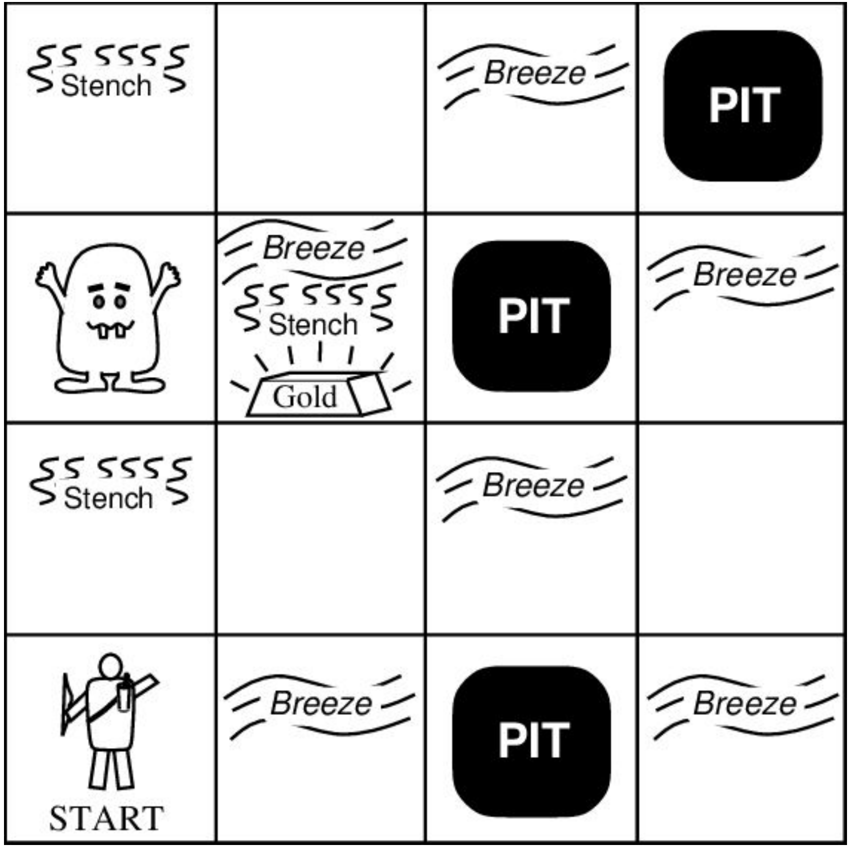
Ali Akbar Septiandri
@aliakbars

Did you get into IT because you wanted to be a game dev? 👨🏻‍💻

Anyone familiar with this image?

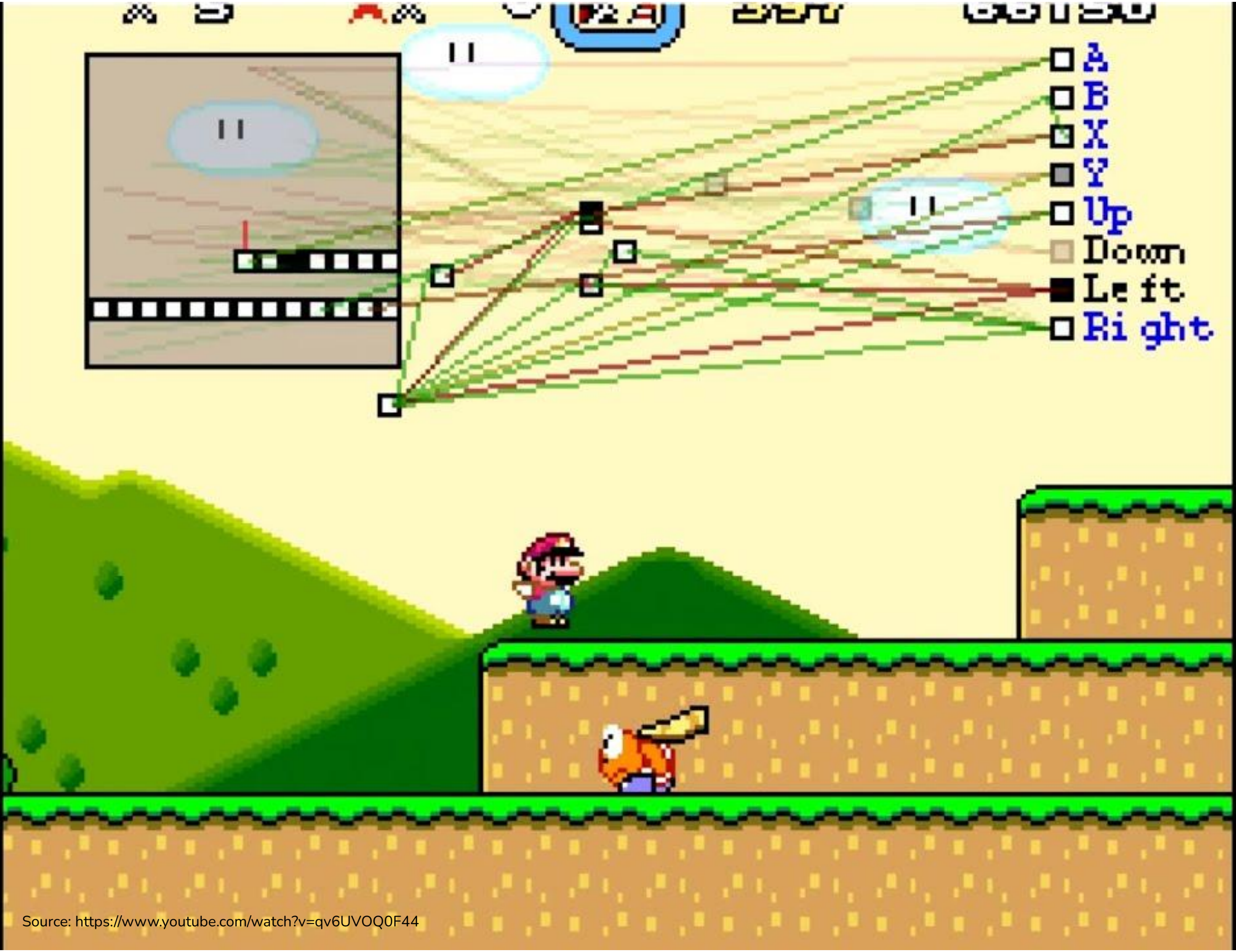


What about this one?

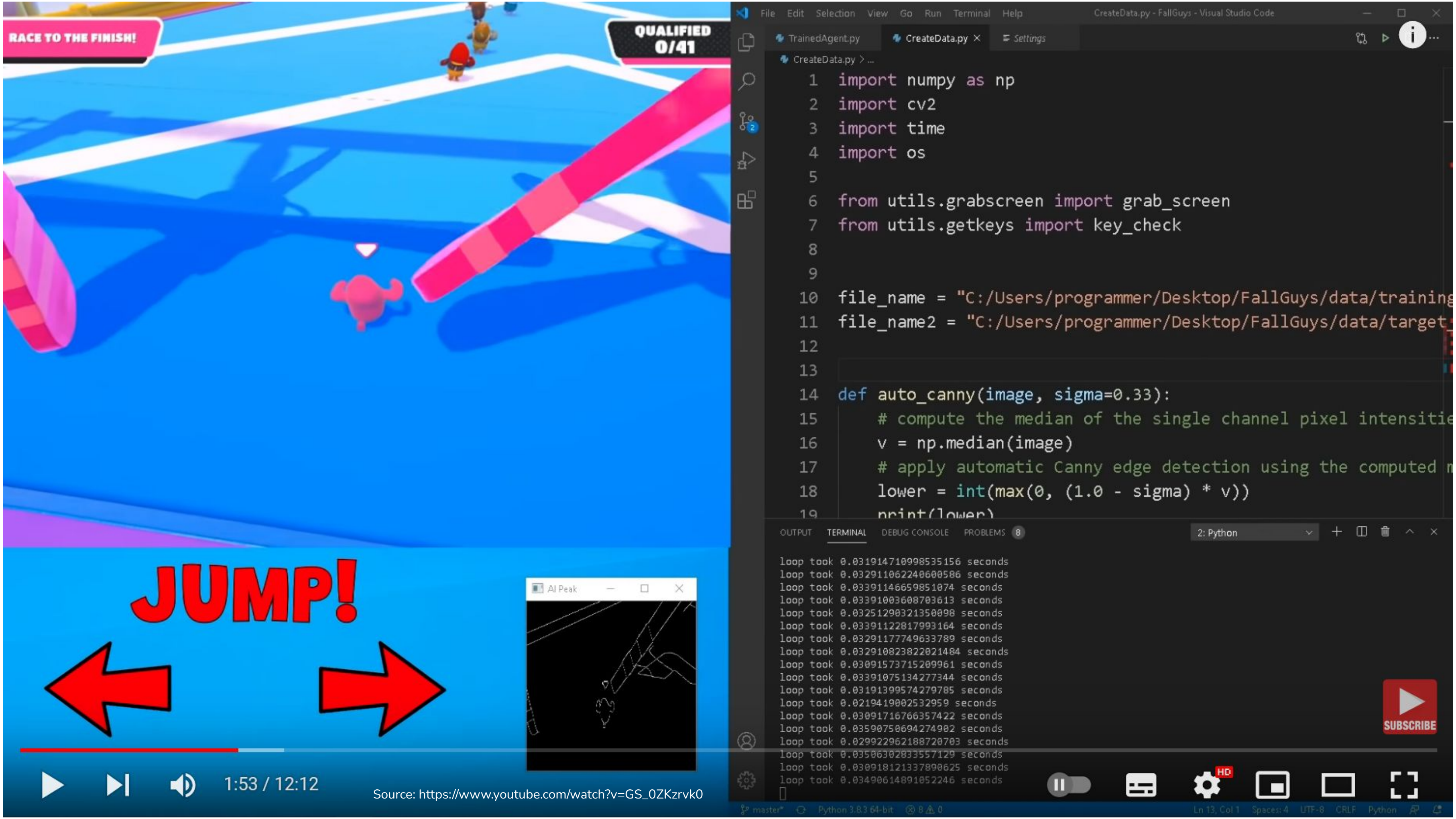


$$V_\pi^{(t)}(s) \leftarrow \sum_{s'} T(s, \pi(s), s')[R(s, \pi(s), s') + \gamma V_\pi^{(t-1)}(s')]$$

What about this? 🧑‍🏫



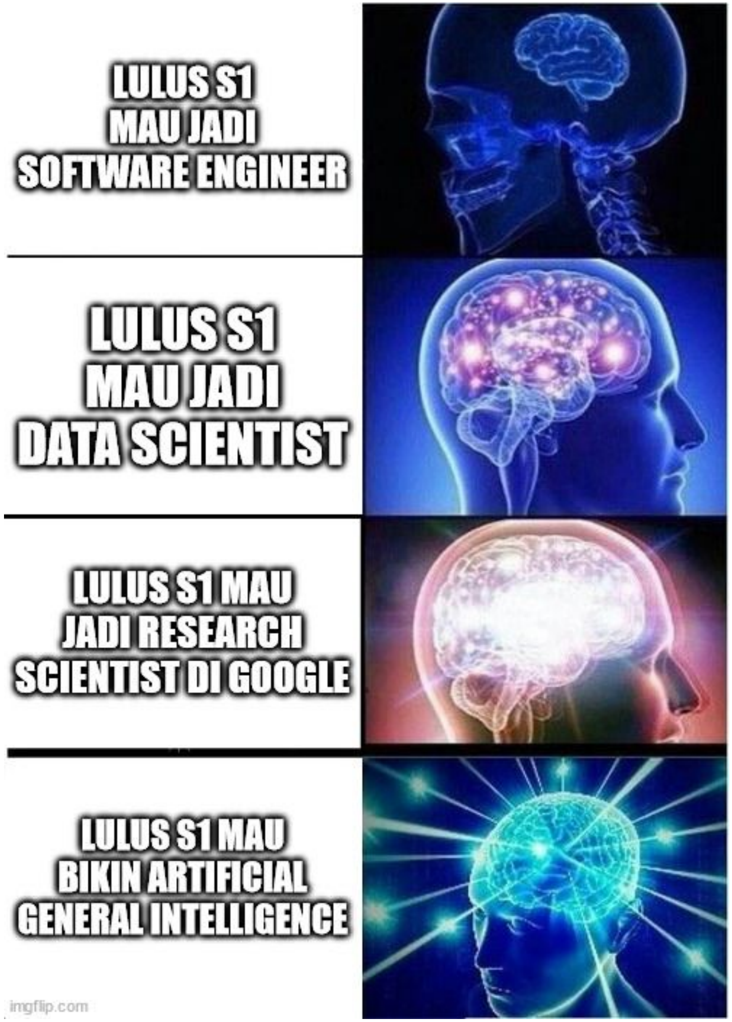
Source: https://www.youtube.com/watch?v=qv6UVOQ0F44



Source: https://www.youtube.com/watch?v=GS_0ZKzrvk0

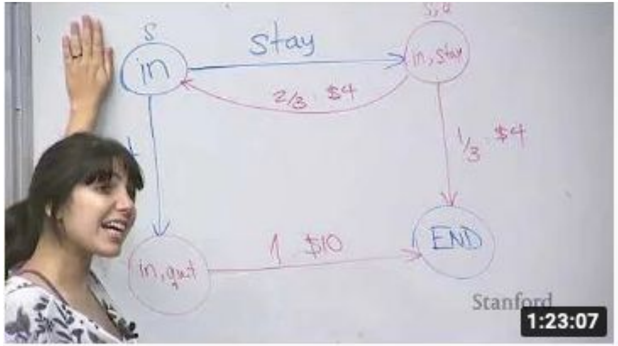
Source: https://www.popularmechanics.com/technology/a19844/googles-alphago-ai-wins-first-round-against-go-champion/


GUIDED MICRO-PROJECTILES
ROCKETS
97% FULL
ROCKETS
97% FULL
ROCKETS
97% FULL
18
15
630


LULUS S1
MAU JADI
SOFTWARE ENGINEER
LULUS S1
MAU JADI
DATA SCIENTIST
LULUS S1 MAU
JADI RESEARCH
SCIENTIST DI GOOGLE
LULUS S1 MAU
BIKIN ARTIFICIAL
GENERAL INTELLIGENCE
imgflip.com

A Whirlwind Tour of Reinforcement Learning

🌲 Stanford CS221: Artificial Intelligence



Lecture 7: Markov Decision Processes - Value Iteration | Stanford CS221: AI (Autumn 2019)

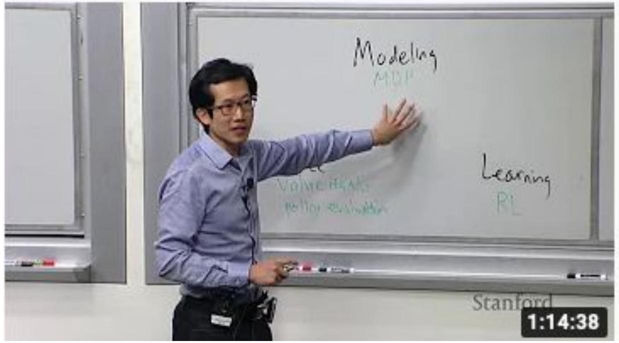
91K views • 1 year ago

stanfordonline

For more information about Stanford's Artificial Intelligence professional and graduate programs, visit: https://stanford.io/3pUNqG7 ...

8:08 Um, and in the middle, I'm going to talk about **policy evaluation**, which is not an inference algorithm but it's kind of a step toward...

Subtitles



Lecture 8: Markov Decision Processes - Reinforcement Learning | Stanford CS221: AI (Autumn 2019)

26K views • 1 year ago

stanfordonline

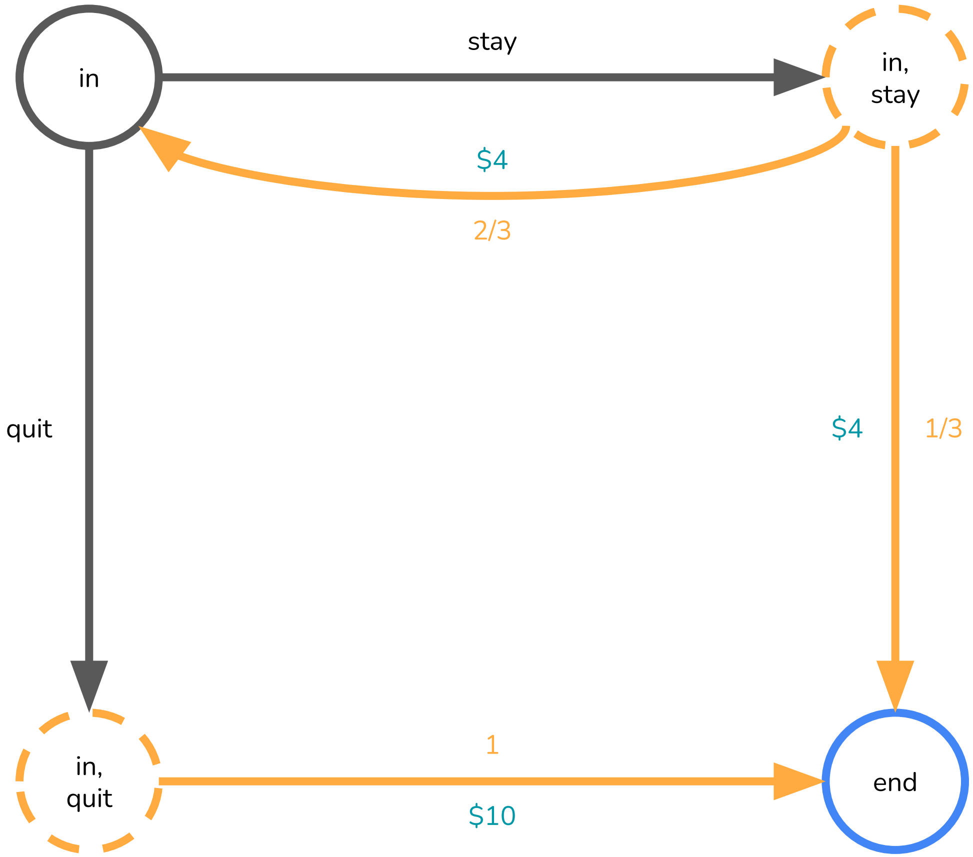
For more information about Stanford's Artificial Intelligence professional and graduate programs, visit: https://stanford.io/2Zv1JpK ...

49:00 So in MDPs, we saw that **policy evaluation** allows you to get Q Pi; value iteration get- allows you to get Q opt. And now, we're ...

Subtitles

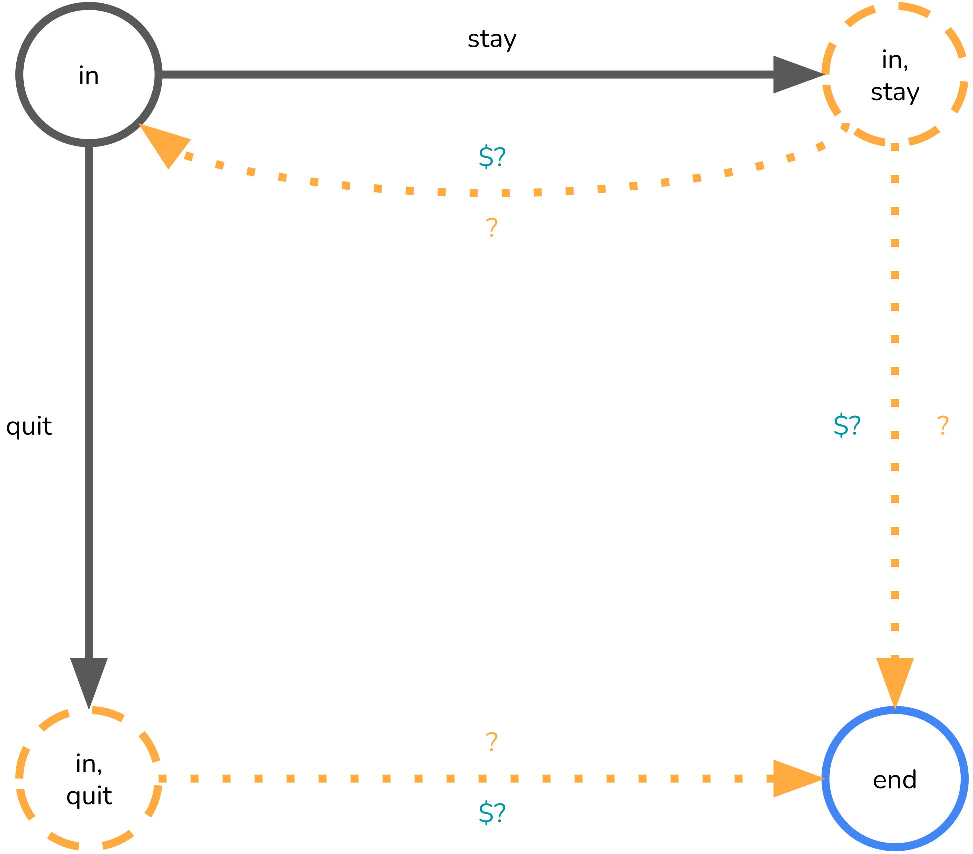
Markov Decision Process

- States
- $s_{start} \in$ States
- Actions(s)
- T(s, a, s')
- Reward(s, a, s')
- IsEnd(s)
- $0 \leq \gamma \leq 1$



Reinforcement Learning

- States
- $s_{start} \in$ States
- Actions(s)
-
-
- IsEnd(s)
- $0 \leq \gamma \leq 1$



$$V_\pi^{(t)}(s) \leftarrow \sum_{s'} \boxed{T(s, \pi(s), s')}\boxed{[R(s, \pi(s), s')} + \gamma V_\pi^{(t-1)}(s')]$$

Model-based
Monte Carlo 💡

Do we have to explore
all states?

SARSA & Q-Learning 💡

Evaluation vs Iteration

SARSA

On each (s, a, r, s’, a’):

$$\hat{Q}_\pi(s,a) \leftarrow (1-\eta)\hat{Q}_\pi(s,a) + \eta(r + \gamma\hat{Q}_\pi(s',a'))$$

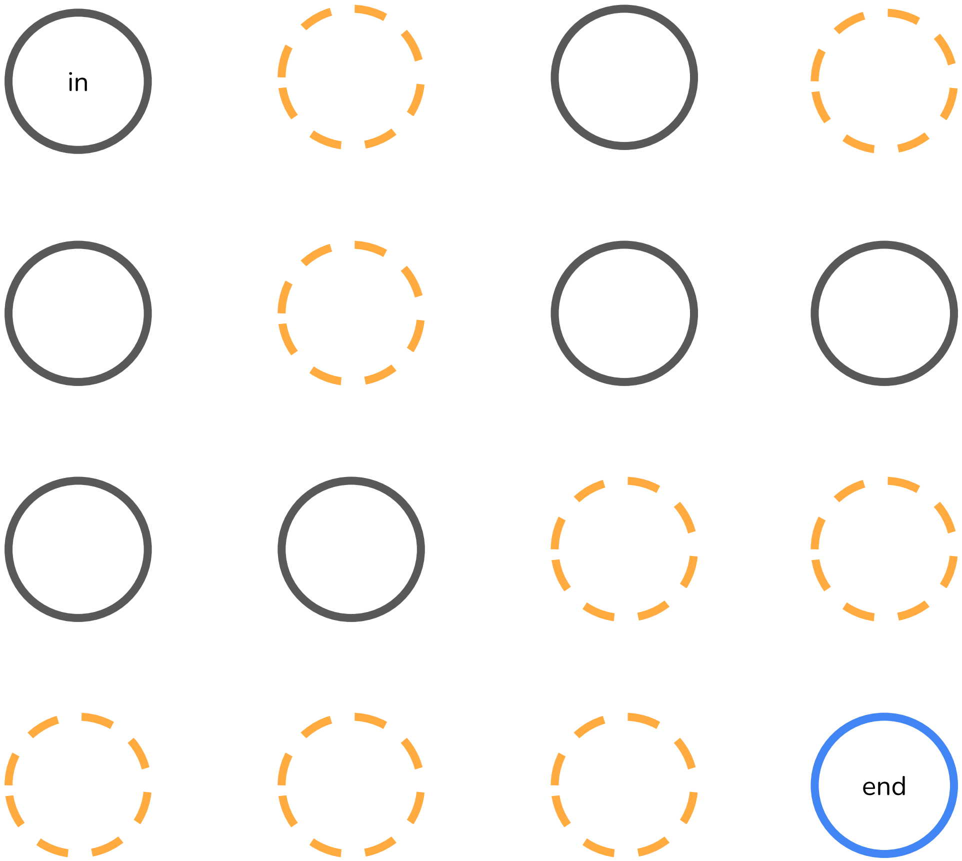
Q-learning

On each (s, a, r, s’):

$$\hat{Q}_{\text{opt}}(s,a) \leftarrow (1-\eta)\hat{Q}_{\text{opt}}(s,a) + \eta(r + \gamma \max_{a' \in \text{Actions}(s')} \hat{Q}_{\text{opt}}(s',a'))]$$

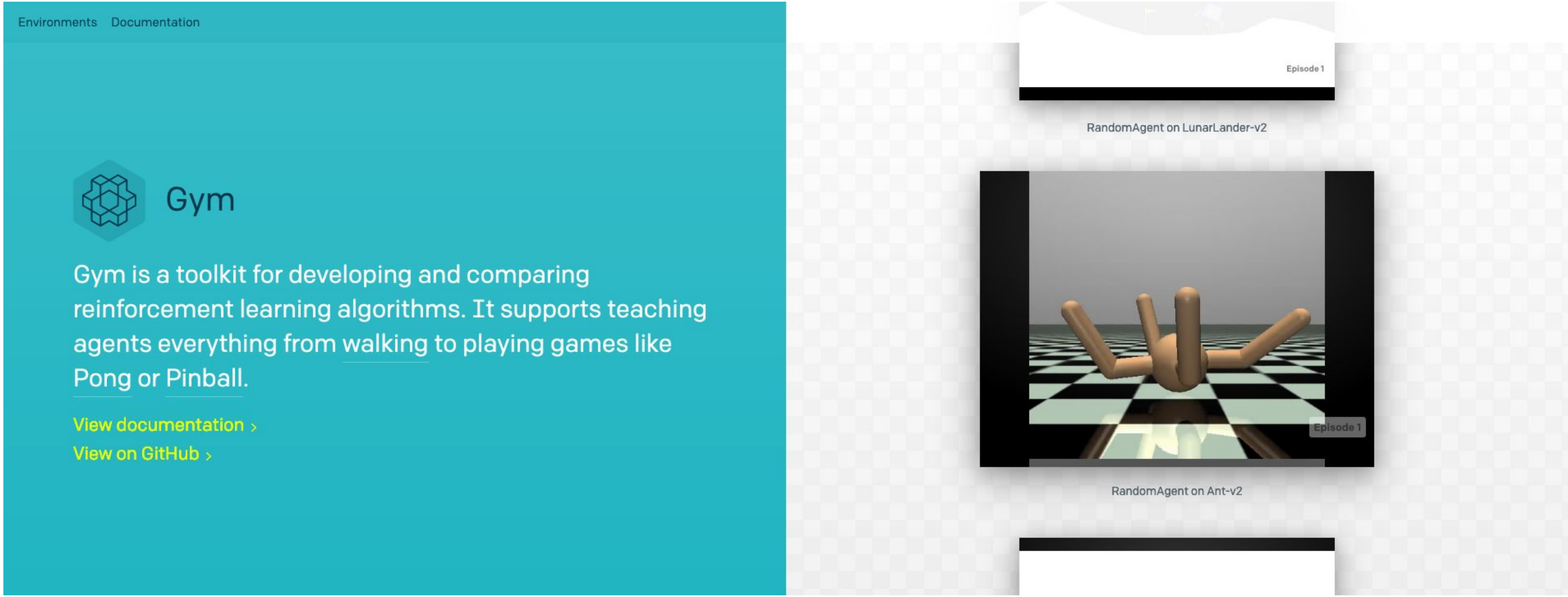
Reinforcement Learning

- States
- $s_{start} \in$ States
- Actions(s)
-
-
- IsEnd(s)
- $0 \leq \gamma \leq 1$



Unlike classical ML, you don't really have scikit-learn and ImageNet / Kaggle

OpenAI Gym


Environments
Documentation
Gym
Gym is a toolkit for developing and comparing reinforcement learning algorithms. It supports teaching agents everything from walking to playing games like Pong or Pinball.
View documentation ›
View on GitHub ›
Episode 1
RandomAgent on LunarLander-v2
Episode 1
RandomAgent on Ant-v2

Environments Documentation

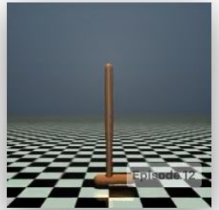
- Algorithms
- Atari
- Box2D
- Classic control
- **MuJoCo**
- Robotics
- Toy text EASY
- Third party environments

MuJoCo

Continuous control tasks, running in a fast physics simulator.

Ant-v2
Make a 3D four-legged robot walk.

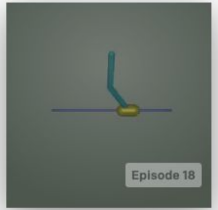
HalfCheetah-v2
Make a 2D cheetah robot run.



Hopper-v2
Make a 2D robot hop.

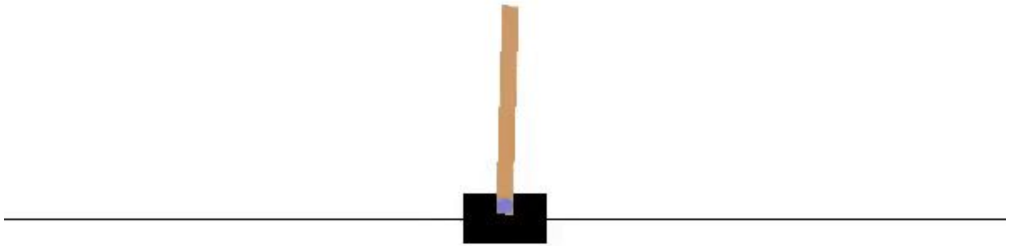
Humanoid-v2
Make a 3D two-legged robot walk.

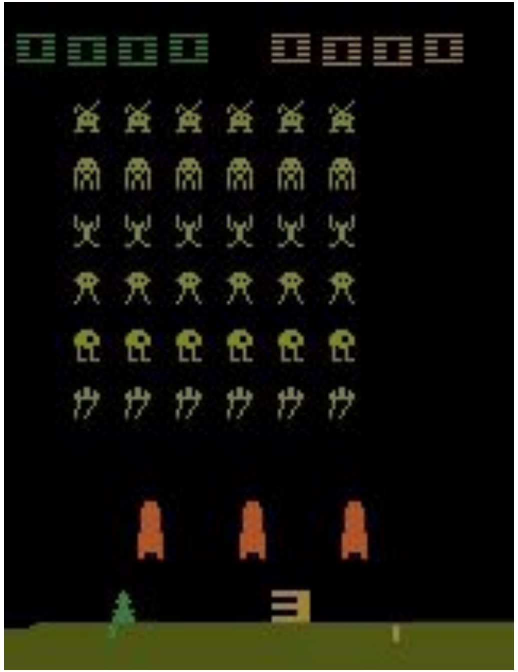
HumanoidStandup-v2
Make a 3D two-legged robot standup.



InvertedDoublePendulum-v2
Balance a pole on a pole on

Examples





```python
import gym
env = gym.make('CartPole-v0')
for i_episode in range(20):
    observation = env.reset()
    for t in range(100):
        env.render()
        print(observation)
        action = env.action_space.sample()
        observation, reward, done, info = env.step(action)
        if done:
            print("Episode finished after {} timesteps".format(t+1))
            break
env.close()
```

Demo

Evaluation vs Iteration

SARSA

On each (s, a, r, s', a'):

$$\hat{Q}_\pi(s,a) \leftarrow (1-\eta)\hat{Q}_\pi(s,a) + \eta(r + \gamma\hat{Q}_\pi(s',a'))$$

Q-learning

On each (s, a, r, s'):

$$\hat{Q}_{\text{opt}}(s,a) \leftarrow (1-\eta)\hat{Q}_{\text{opt}}(s,a) + \eta(r + \gamma \max_{a' \in \text{Actions}(s')} \hat{Q}_{\text{opt}}(s',a'))]$$

Function approximation 💡

$$\hat{Q}_{\text{opt}}(s, a; \mathbf{w}) = \mathbf{w} \cdot \phi(s, a)$$

Q can be approximated as weights x features

$$\mathbf{w} \leftarrow \mathbf{w} - \eta[\underbrace{\hat{Q}_{\text{opt}}(s, a; \mathbf{w})}_{\text{prediction}} - \underbrace{(r + \gamma \hat{V}_{\text{opt}}(s'))}_{\text{target}}]\phi(s, a)$$

On each (s, a, r, s'), fix the weights

Combining with Deep Learning 💡

RL Applications

- Game playing 🕹️🎮
- Multi-armed bandits 🎰
- Dialogue systems 💬
- ...

Learning Resources

1. Stanford CS221: Artificial Intelligence
2. DeepMind/UCL: Reinforcement Learning by David Silver
3. UC Berkeley CS285: Deep Reinforcement Learning by Sergey Levine
4. Reinforcement Learning: An Introduction by Sutton & Barto

Conclusions

1. RL is often overlooked in ML syllabi because of the hassles 👀
2. OpenAI Gym can help you teach/learn RL – from Q-learning to DQN! 🏋🏻💪🏻
3. Learning by doing is the best 👨🏻‍💻🐱💻👩🏻‍💻

Thank you

@aliakbars