Lab 11: Reinforcement Learning

With a focus on Homework 8

Harvard IACS

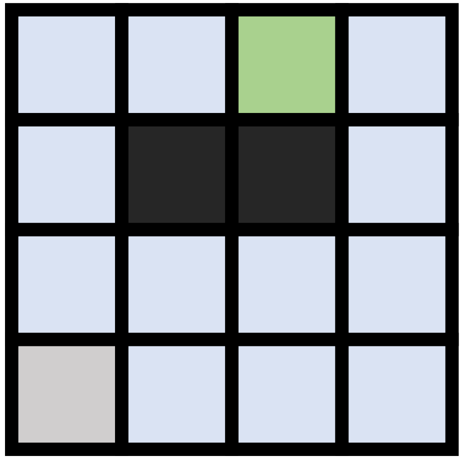
CS109B

Chris Tanner, Pavlos Protopapas, Mark Glickman

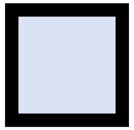
- Agent

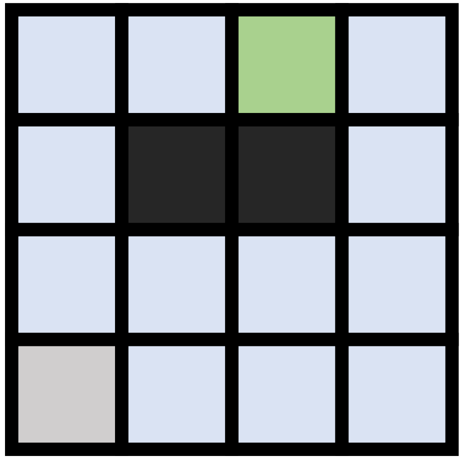
The thing that operates
within the environment

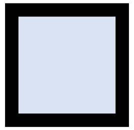
RL Environment



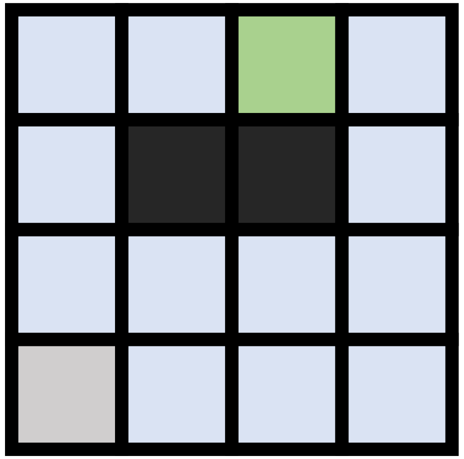
RL Environment

- Agent  The thing that operates within the environment
- States  Static representations that define the current environment



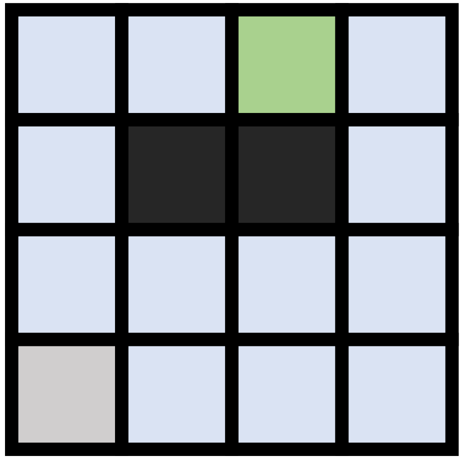
- Agent  The thing that operates within the environment
- States  Static representations that define the current environment
- Actions  An agent's operation that takes him/her from state **s** to state **s'**

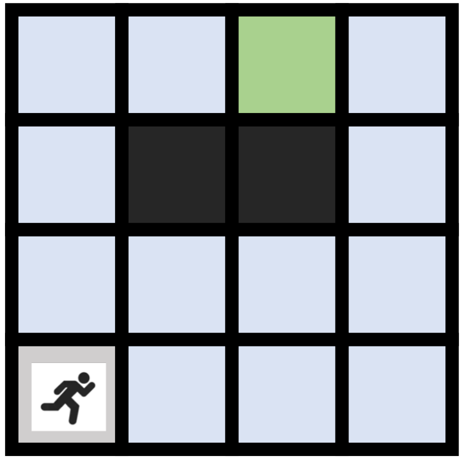
RL Environment

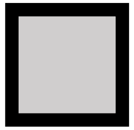


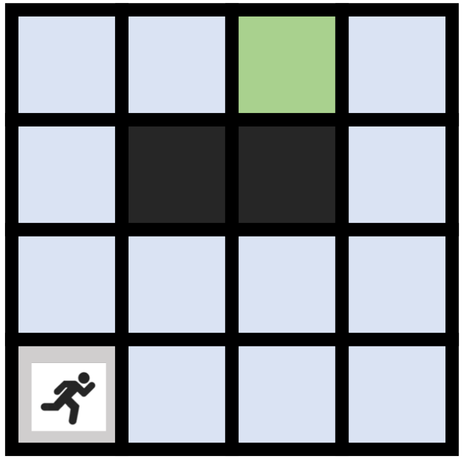
RL Environment

- **Agent** — The thing that operates within the environment
- **States** — Static representations that define the current environment
- **Actions** — An agent's operation that takes him/her from state **s** to state **s'**
- **Reward** — A real-valued # that represents the goodness for the agent's being in a given state **s**



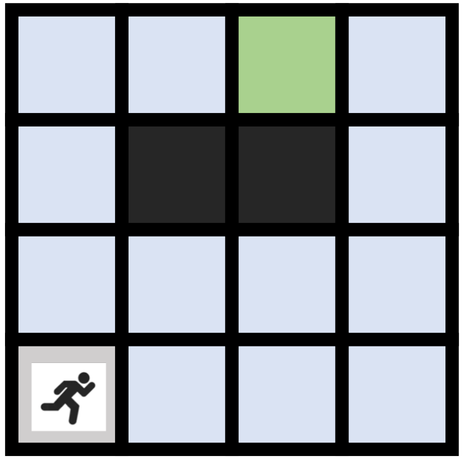


The agent  starts in an initial state **s** 



The agent starts in an initial state **s**

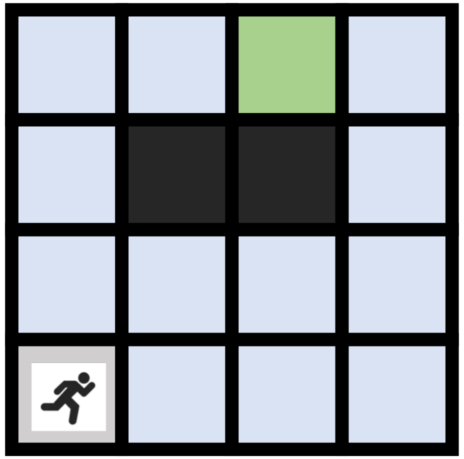
She performs an action **a** → and becomes in state **s′**



The agent starts in an initial state **s**

She performs an action **a** → and becomes in state **s′**

Being in each state **s′** yields a reward **r** (e.g, 3.6)

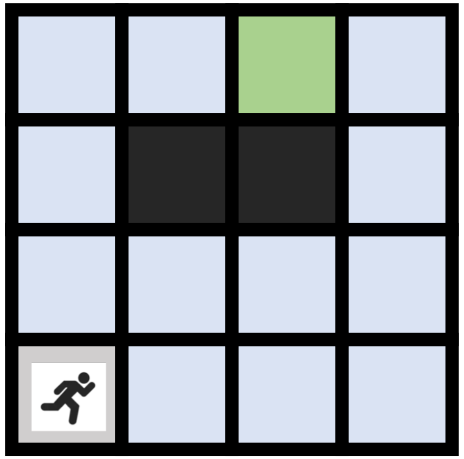


The agent  starts in an initial state **s**

She performs an action **a** → and becomes in state **s′**

Being in each state **s′** yields a reward **r** (e.g, 3.6)

How do we determine how to move from state-to-state so as to receive maximum reward **r**?



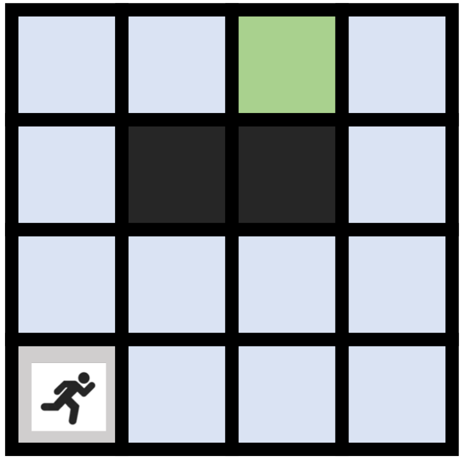
The agent starts in an initial state **s**

Sh

Be

That’s the entire crux of reinforcement learning!

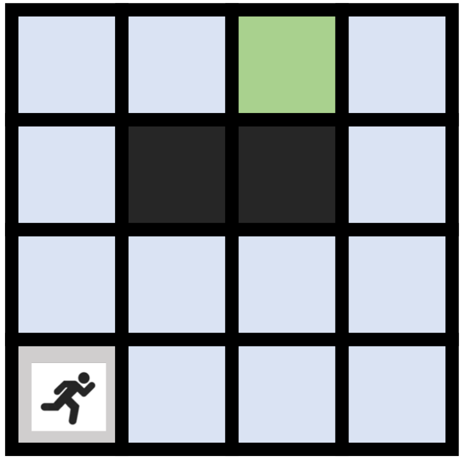
How do we determine how to move from state-to-state so as to receive maximum reward **r**?



Given a state **s**, and an action **a**, estimate the reward **r**.

A policy $\pi(\mathbf{s})$ takes a state **s** and executes an action **a**

$$\pi(\mathbf{s}) \rightarrow \boldsymbol{a}$$



So, there can be many policies $\pi_1, \pi_2, \dots, \pi_n$

(some better than others).

which policy to execute?

How do we determine ~~how to move from state-to-state so as to receive maximum reward r~~?

The one that gives us the highest reward!
Let's estimate each policy π_i via a "value-function" $\boldsymbol{v}_\pi(s)$, where s is our starting state

In class, we saw that:

$$\boldsymbol{v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right] = \mathbb{E}_\pi\left[\sum_{k=0} \gamma^k R_{t+k+1} \,|s\right]}$$

$$\boldsymbol{= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]}$$

In class, we saw that:

$$\boldsymbol{v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right] = \mathbb{E}_\pi\left[\sum_{k=0} \gamma^k R_{t+k+1} \,|s\right]}$$

The policy we're interested in

$$\boldsymbol{= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]}$$

In class, we saw that:

Our starting state s

$$v_\pi(s) = \mathbb{E}_\pi \left[G_t | S_t = s\right] = \mathbb{E}_\pi \left[\sum_{k=0} \gamma^k R_{t+k+1} \,|s\right]$$

The policy we're interested in

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a)[r + \gamma v_\pi(s')]$$

In class, we saw that:

Our starting state s

Cumulative reward

$$v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right] = \mathbb{E}_\pi\left[\sum_{k=0} \gamma^k R_{t+k+1} | s\right]$$

The policy we're interested in

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a)[r + \gamma v_\pi(s')]$$

In class, we saw that:

Our starting state s

Cumulative reward

Unadjusted reward

$$v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right] = \mathbb{E}_\pi\left[\sum_{k=0} \gamma^k R_{t+k+1} \,|s\right]$$

The policy we're interested in

$$= \sum_a \pi(a|s) \sum_{s\prime} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

In class, we saw that:

Weakened based on how far into the future it is

Our starting state s

Cumulative reward

Unadjusted reward

$$v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right] = \mathbb{E}_\pi\left[\sum_{k=0} \gamma^k R_{t+k+1} \,|s\right]$$

The policy we're interested in

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

HW #1.1

state-to-state transitions **T**

reward **R**

T=

R=

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

HW #1.2

- Estimate v_π
- The above equation is general and works for stochastic situations.
- We have fixed state-transitions and rewards though (we can make our life easier).

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

$$= \sum_a \pi(a|s) \sum_{s'} p(s', r|s, a)(R(s, a) + \gamma v_\pi(s'))$$

HW #1.2

- Estimate v_π
- The above equation is general and works for stochastic situations.
- We have fixed state-transitions and rewards though (we can make our life easier).
- What can we define as a constant?

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

$$= \sum_a \pi(a|s) \sum_{s'} p(s', r|s, a)(R(s, a) + \gamma v_\pi(s'))$$

HW #1.2

- As a sanity check, a geometric series is defined to have a sum:

$$a + ar + ar^2 + ar^3 + ar^4 + \cdots = \sum_{k=0}^{\infty} ar^k = \frac{a}{1-r}, \text{ for } |r| < 1.$$

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)[r + \gamma v_\pi(s')]$$

$$= \sum_a \pi(a|s) \sum_{s'} p(s', r|s, a)(R(s, a) + \gamma v_\pi(s'))$$

HW #1.3

- What is a Q-function?

$$v_{\pi}(s) = \sum_{a} \pi(a|s) \sum_{s'} \sum_{r} p(s', r|s, a)[r + \gamma v_{\pi}(s')]$$

$$= \sum_{a} \pi(a|s) \sum_{s'} p(s', r|s, a)(R(s, a) + \gamma v_{\pi}(s'))$$

HW #1.3

- What is a Q-function?