CS 277: Control and Reinforcement Learning

Winter 2021

Lecture 1: Introduction

Roy Fox
Department of Computer Science
Bren School of Information and Computer Sciences
University of California, Irvine

Today's lecture

What is reinforcement learning?

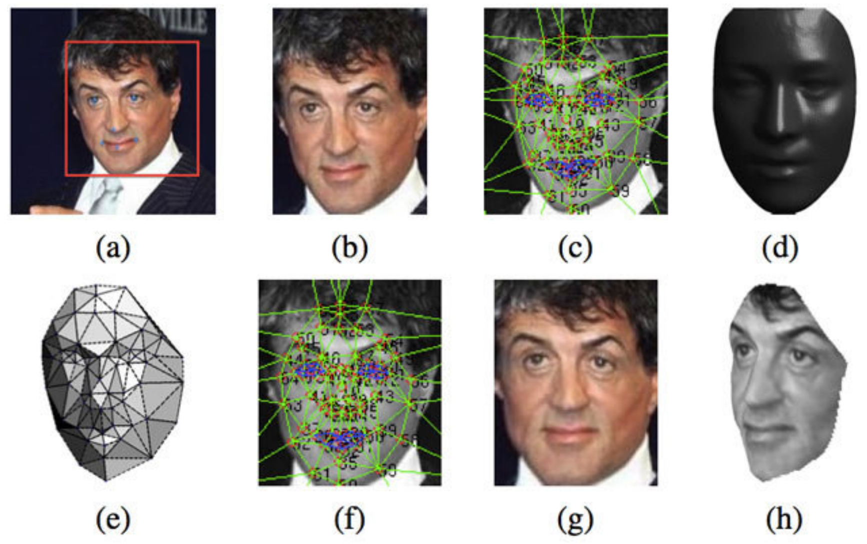
Course logistics

Basic RL concepts

What is machine learning

- Can we build “intelligent” machines? Intelligence = good decision making
- Learning = taking in information to “know” more than you did before
- Machine learning = use data to make better decisions than before [Mitchell 1997]
- ML can help when other AI methods fail:
  - Experts are scarce
  - Rules / logic are hard to specify
  - Search space is too large
  - Models are unknown / hard to specify

Face recognition



Speech synthesis

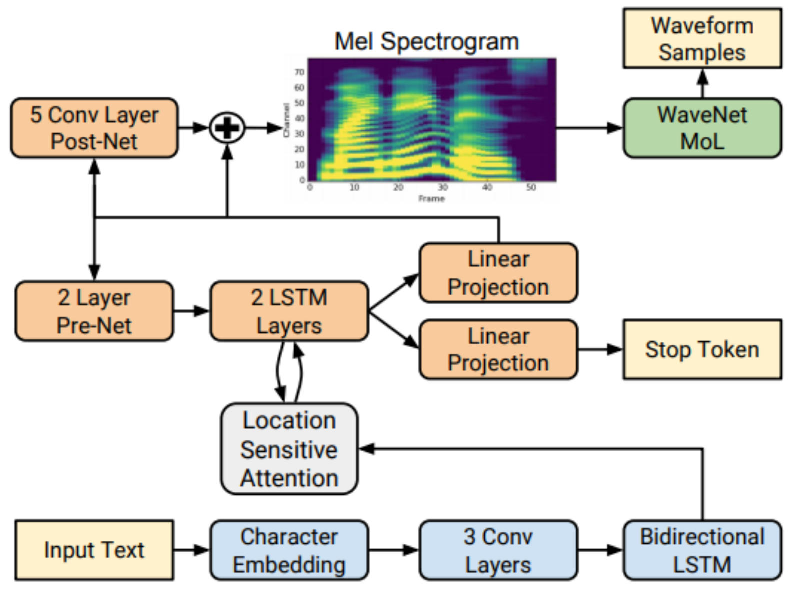
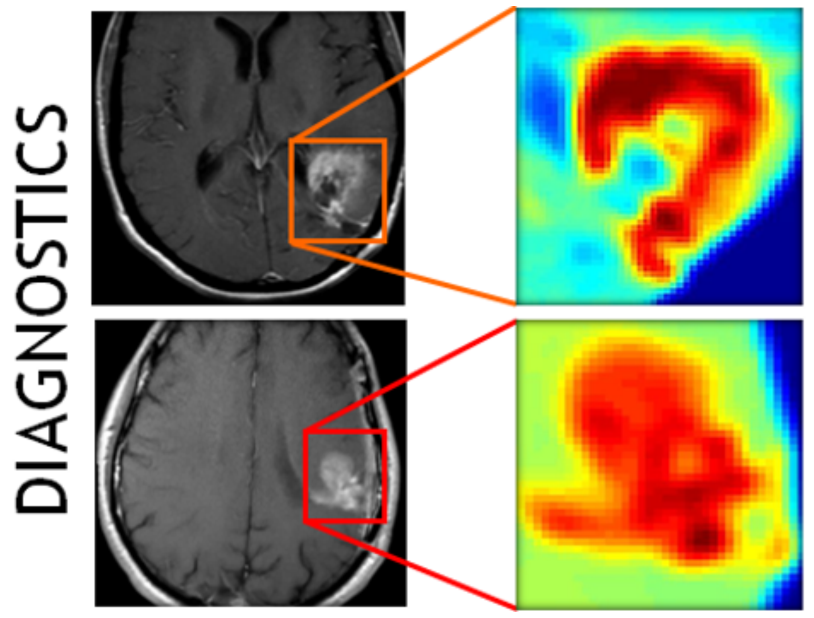
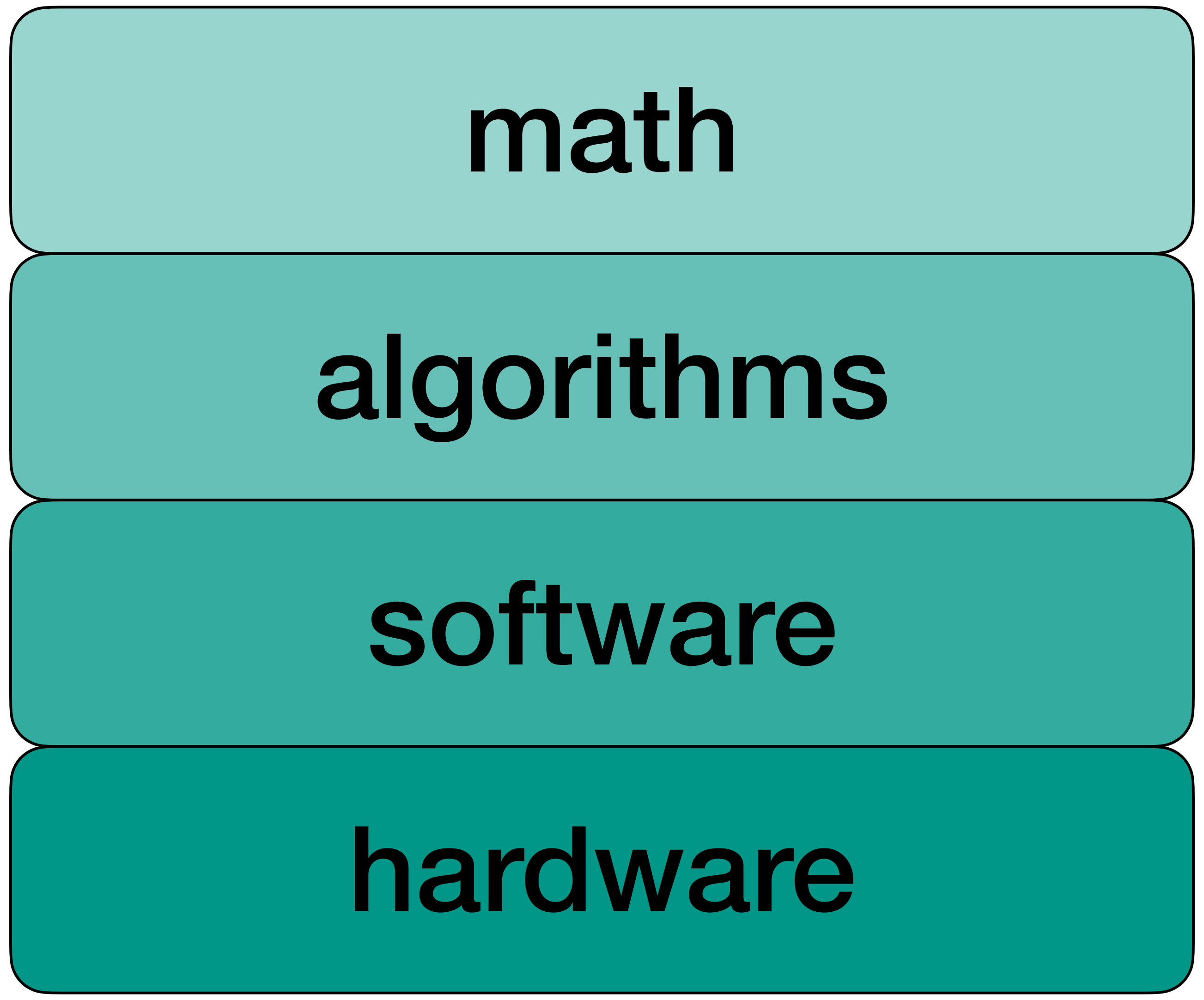


Fig. 1. Block diagram of the Tacotron 2 system architecture.

Medical diagnosis



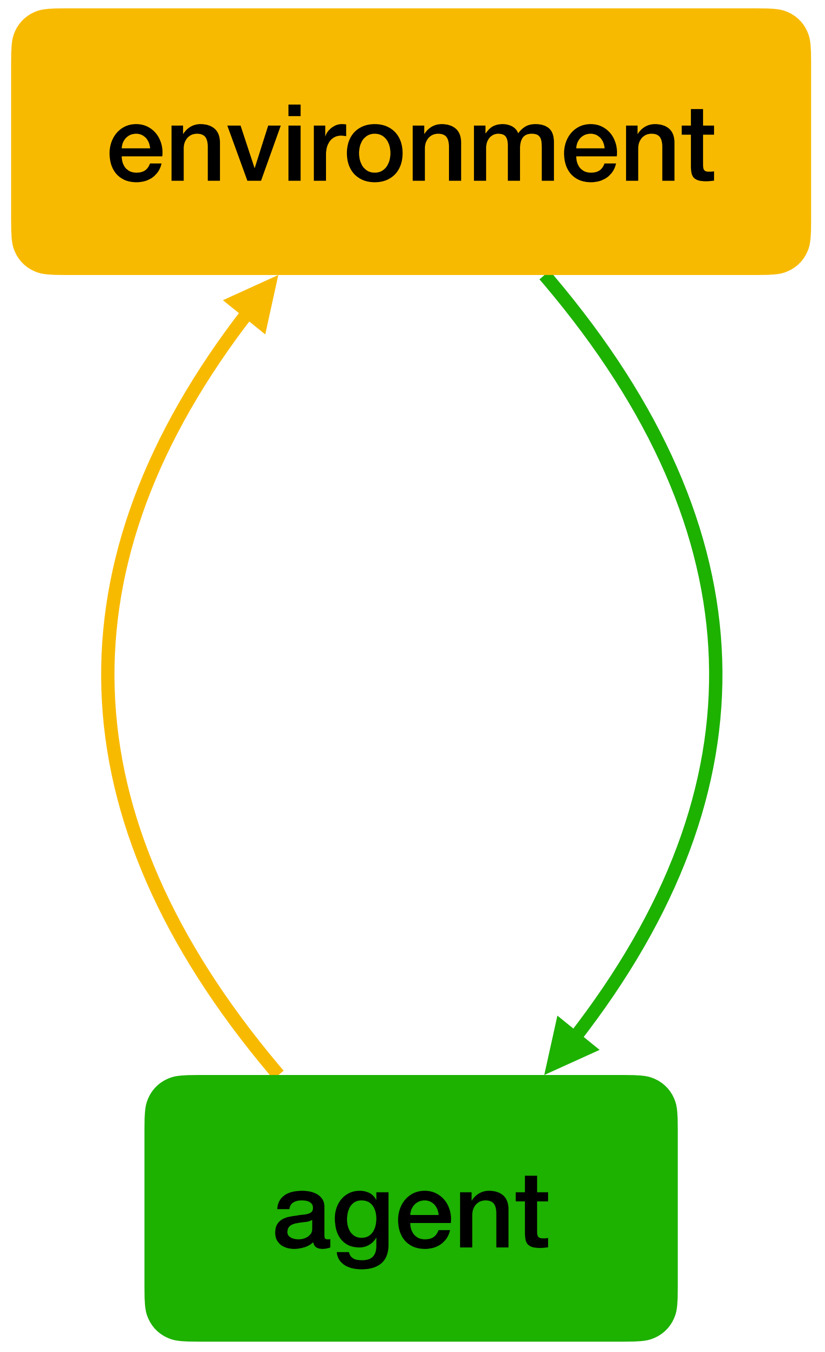
The ML stack



- Math: probability theory, (linear) algebra, computational learning theory
- Algorithms: ML algorithms, optimization, data structures
- Software: ML frameworks, databases, testing, deployment
- Hardware: cloud computing, distributed systems, cyber-physical systems

What is control learning?

- Intelligence appears in interaction with a complex system, not in isolation
  - An agent interacting with an environment
- Control = sequential decision making
  - Sense environment state s
  - Take action a
  - Repeat

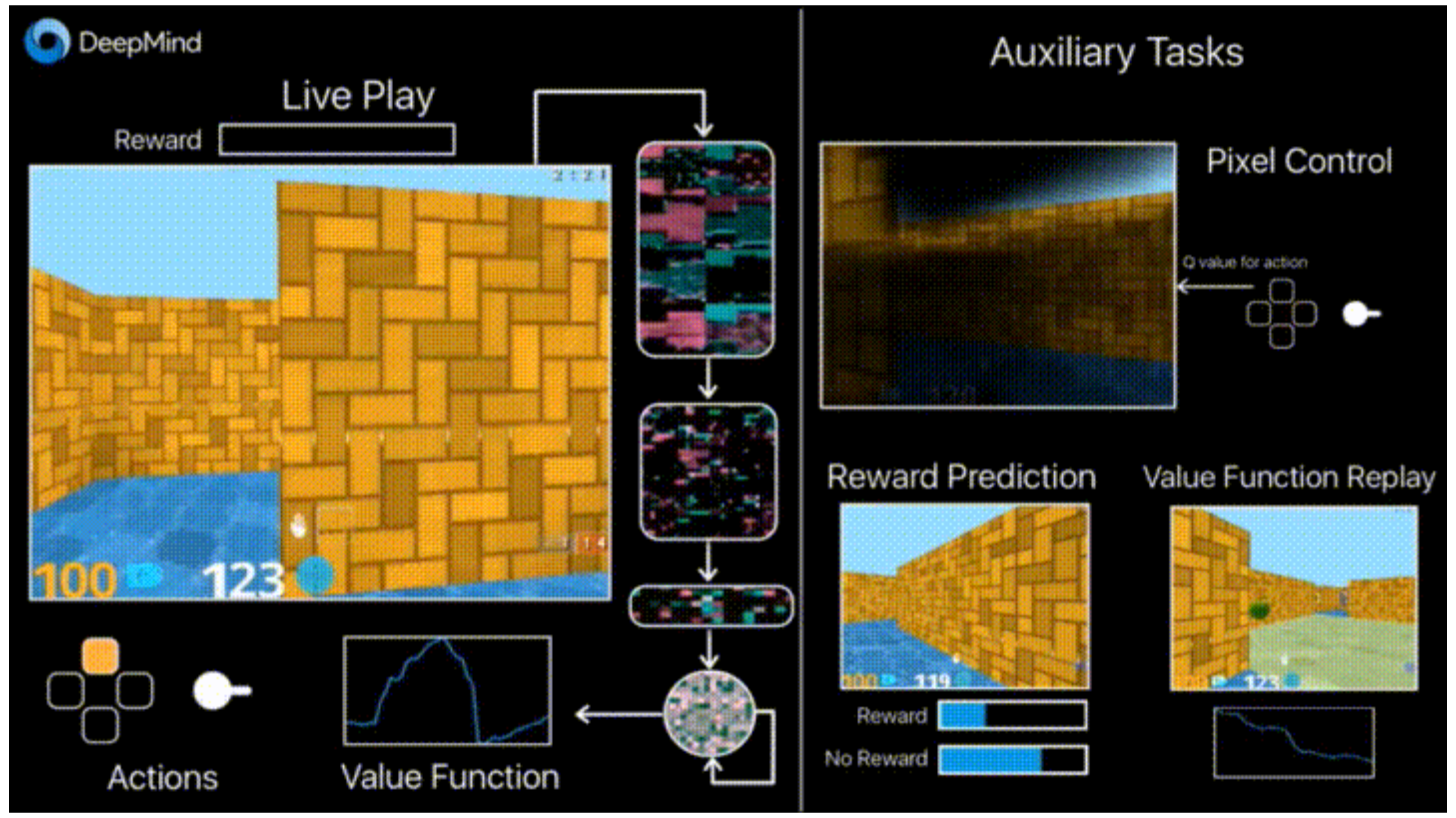


- Success can be measured by matching good actions — imitation learning (IL)
  - Or by accumulating high rewards $r(s, a)$ — reinforcement learning (RL)

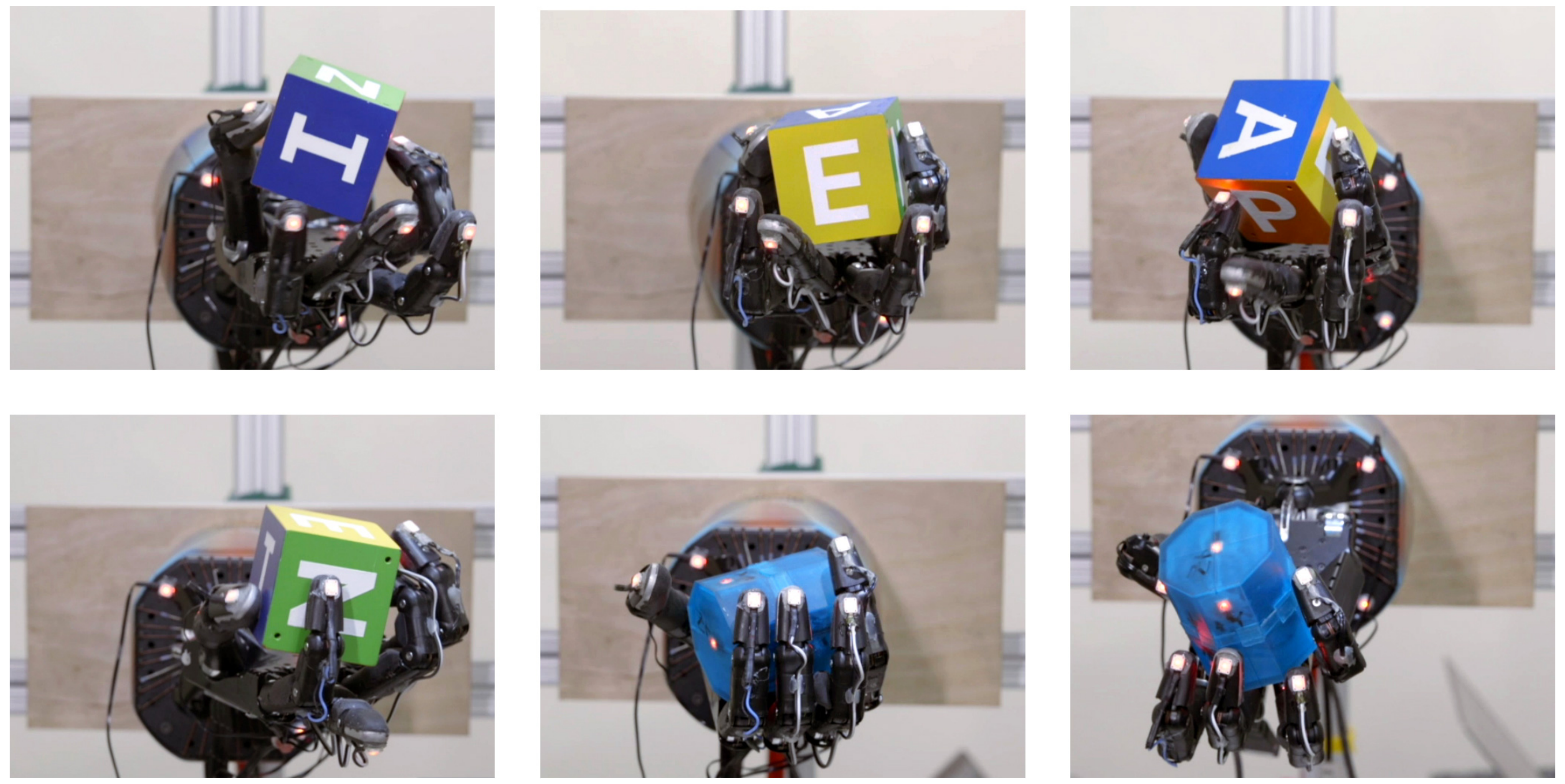
Examples of learned controllers

Gameplay

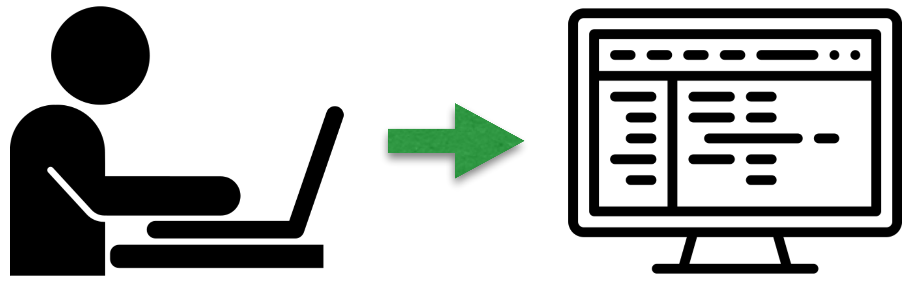
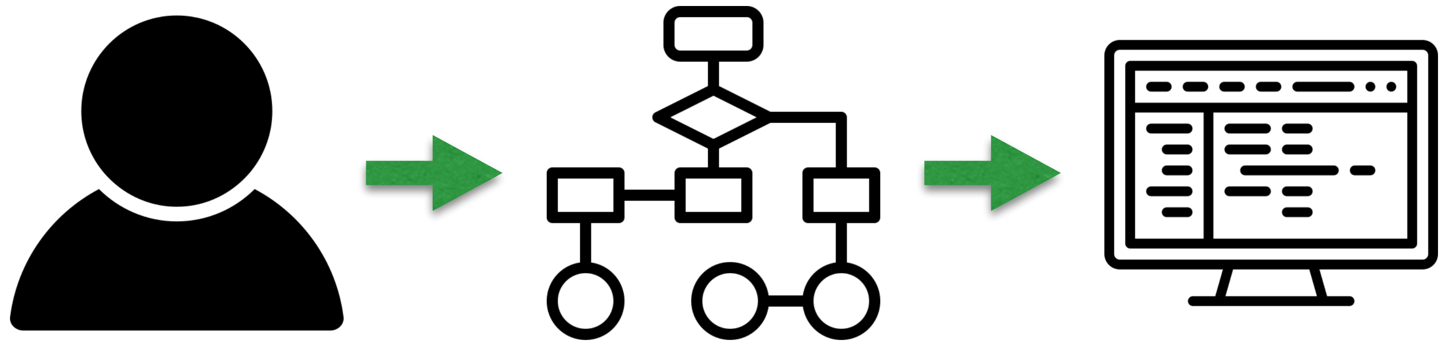
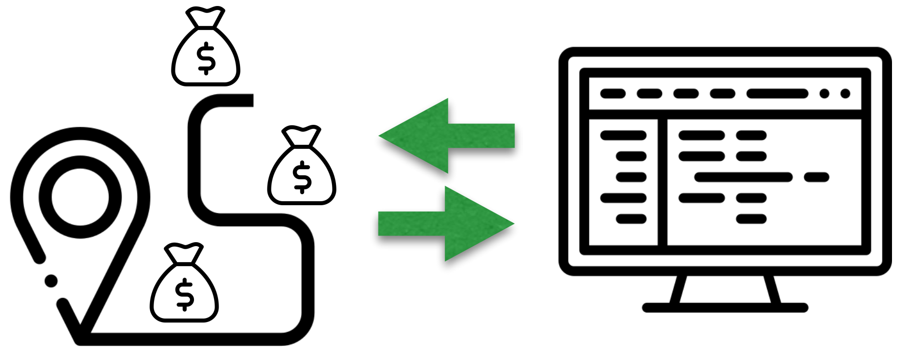
Spacial navigation



Dextrous manipulation



Control preference elicitation

	Explicit	Implicit
"how"	Programming 	Imitation Learning 
"what"	Instruction following 	Reinforcement Learning 

Control learning is ML... but special

- In RL, unlike supervised, no ground truth, only feedback (online learning)
- Exploration = the learner collect data by interaction — very challenging
  - The agent decides on which states to train (active learning) — and test!
  - Cannot avoid train–test mismatch
- Sequential decision making need to be coordinated
  - Optimization space is strewn with local optima
- A good policy may require memory
  - Learning to remember is very challenging

Today's lecture

What is reinforcement learning?

Course logistics

Basic RL concepts

Course logistics

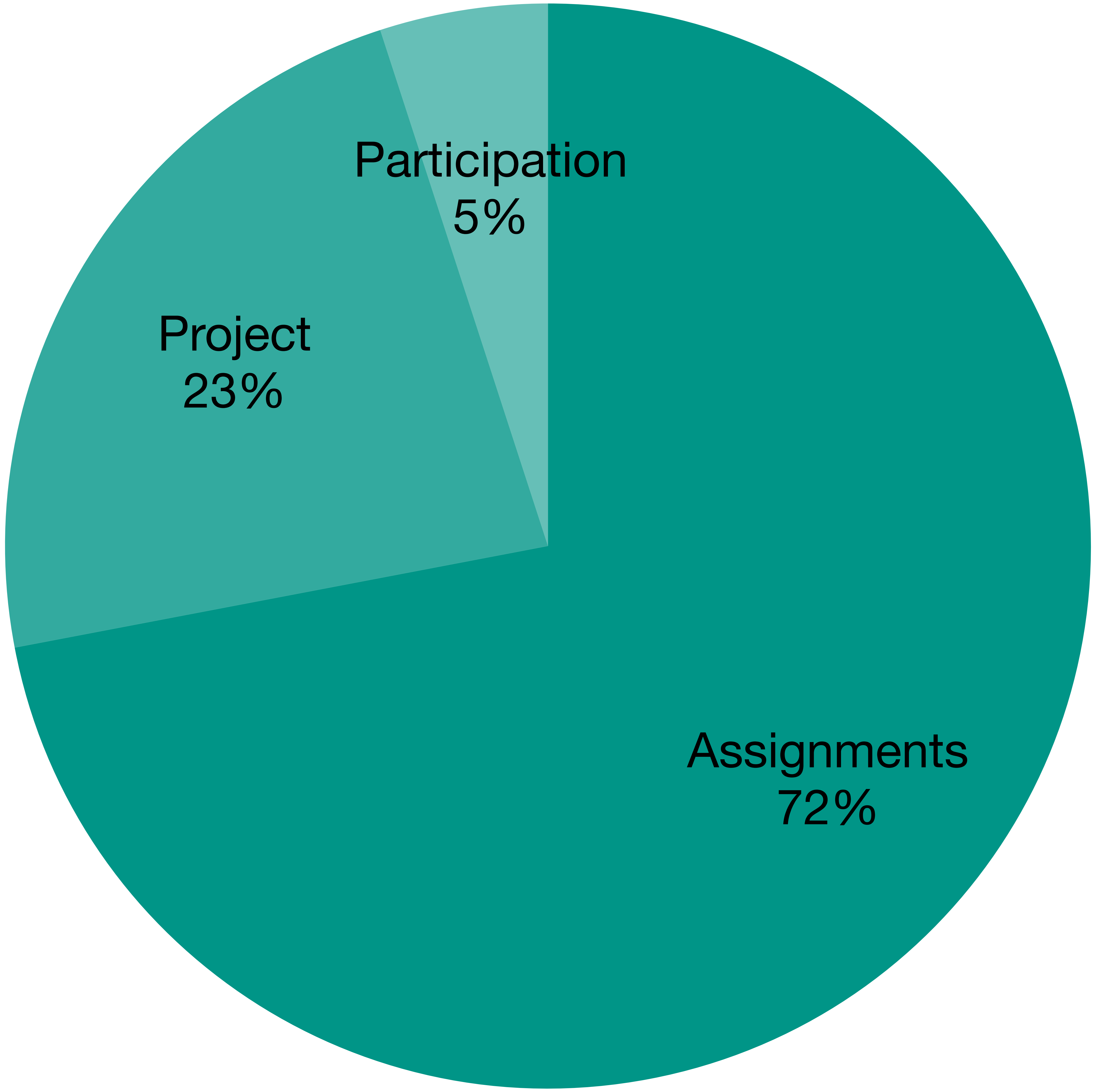
- When: Tuesdays and Thursdays, 5–6:20pm
  - Lectures will be recorded and published afterwards
- Where: https://uci.zoom.us/j/96005379683
- Website: https://royf.org/crs/W21/CS277/ ← Schedule! Resources!
- Forum: https://piazza.com/uci/winter2021/cs277
  - For announcement and questions (no email please!)
- Biweekly assignments: https://www.gradescope.com/courses/221674
- Office hours: https://calendly.com/royfox/office-hours
  - Welcome to schedule 15-min slots and invite friends; give 4 hour notice

Compute resources

- Most assignments should fit on your personal computer
  - Always start by testing your code on a smaller challenge that “should” work
- If more compute resources are required:
  - Campus-wide cluster: https://rcic.uci.edu/hpc3/
  - Google Colab: https://colab.research.google.com/

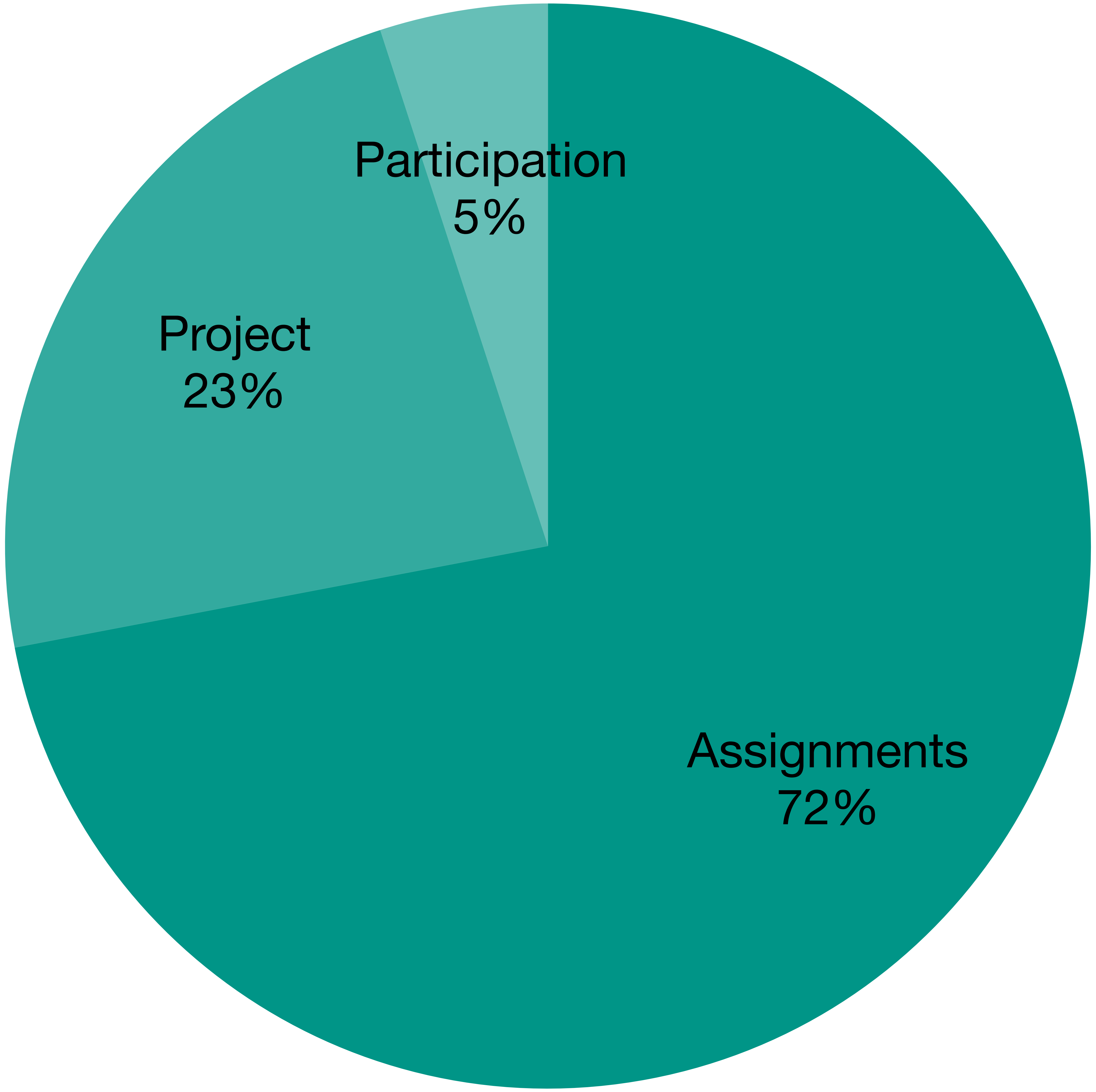
Grading policy

- 5 assignments + project
  - Understand RL theory
  - Apply RL techniques in Python
  - Show your math, code, and results
- Grading:
  - 4 best assignments count for 18% each
  - Project counts for 23%
  - No late submission



Grading: participation

- Forum participation
  - Ask questions if you have any
  - Answer questions if you can
  - Post relevant useful links
  - Upvote useful posts
  - Give private feedback to staff
- Quizzes, surveys, and evaluations
  - Answer polls published on the forum
  - Submit course evaluations



What will it take to do well?

- We'll rely heavily on math: probability theory, linear algebra, calculus
  - I'm here to help, but solid background expected

- You'll need to code well in Python
- Some ideas are challenging — ask early what you don't fully understand
- Help your friends and get help — from me too! — but never cheat

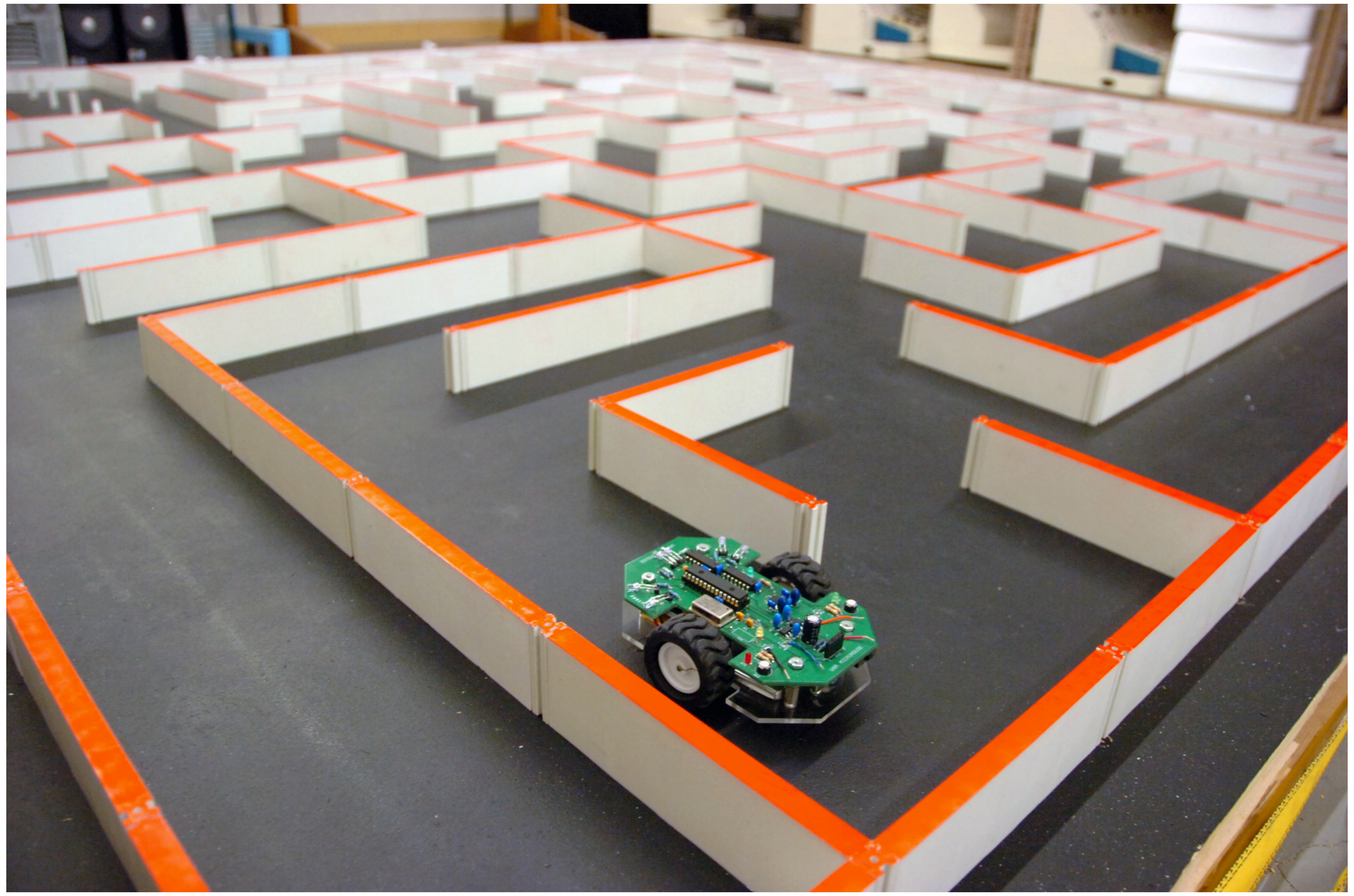
Today's lecture

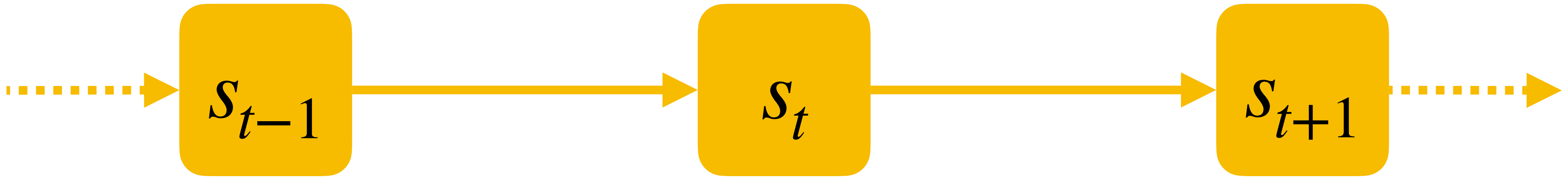
What is reinforcement learning?

Course logistics

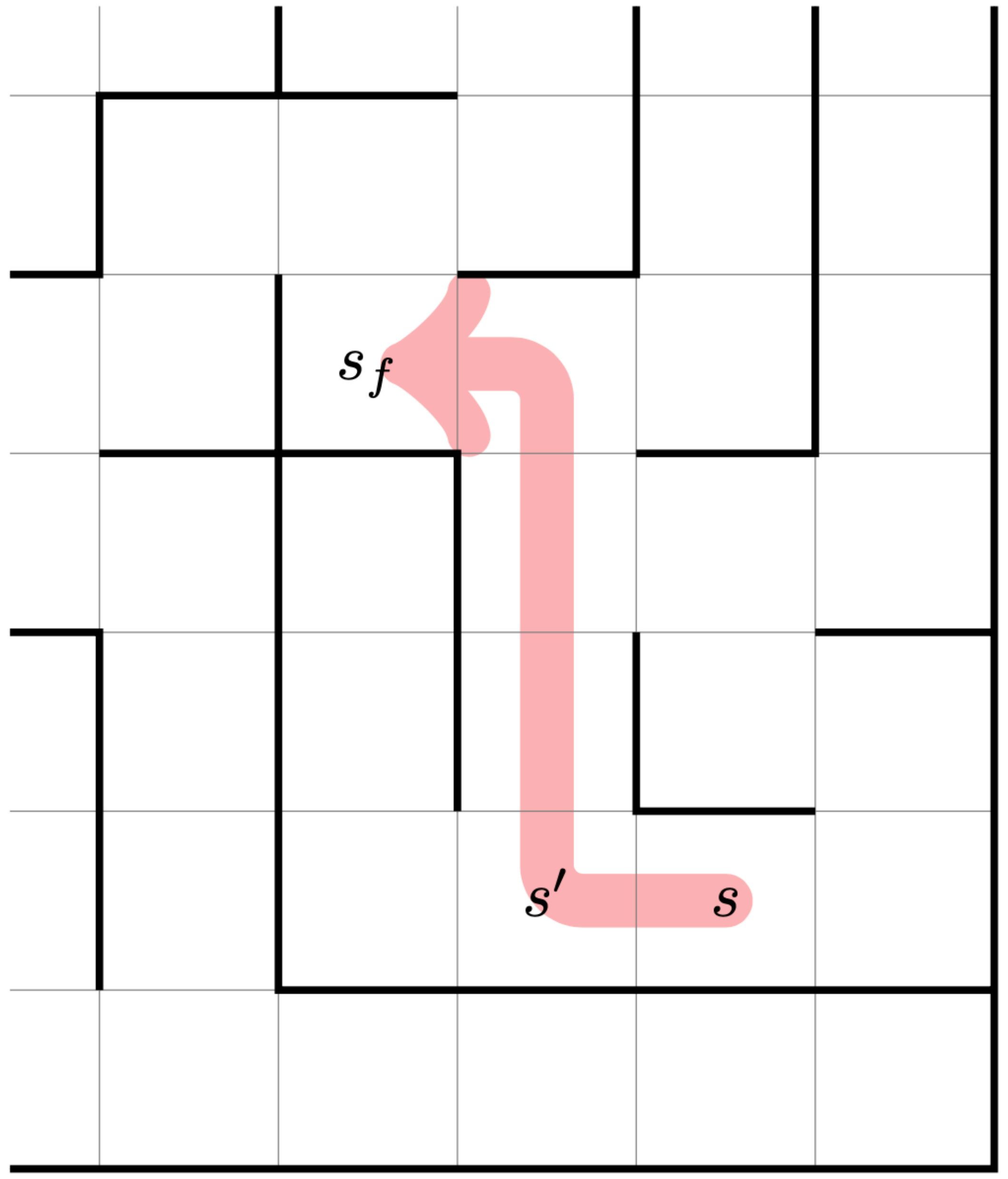
Basic RL concepts

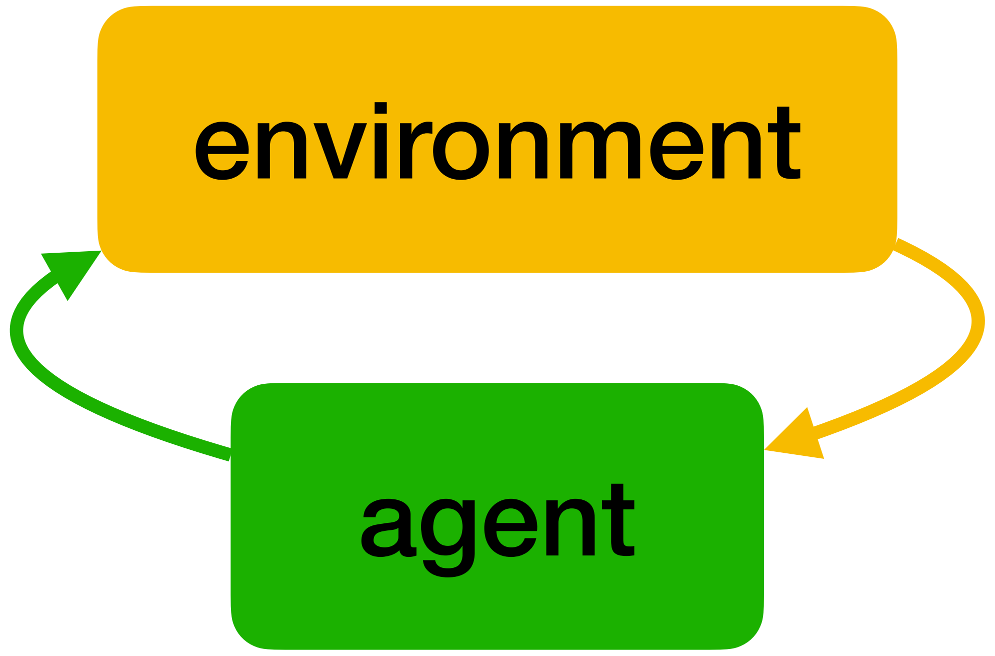
System state

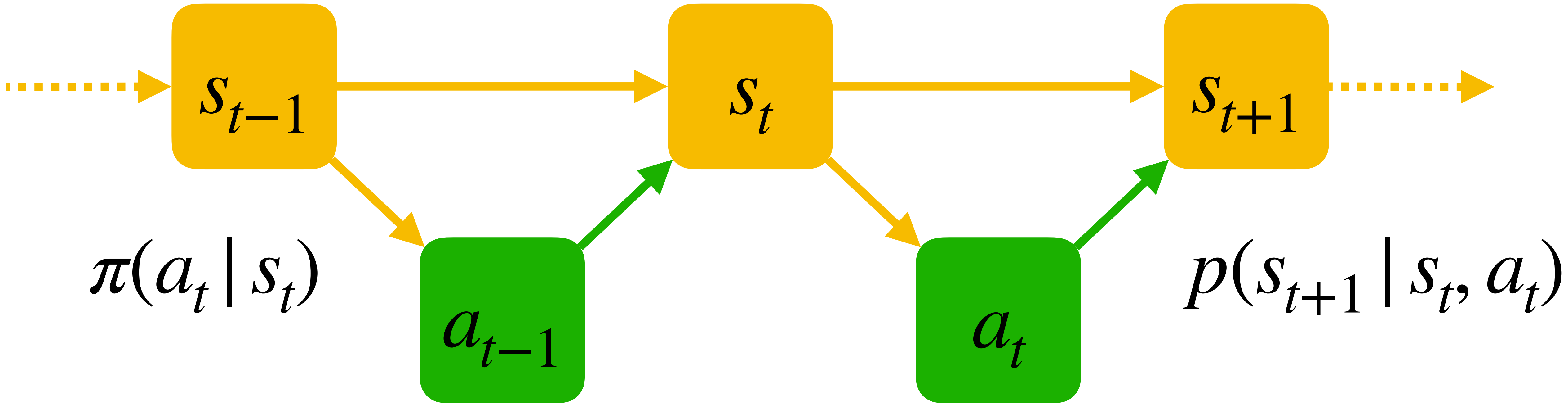




System = agent + environment







Basic RL concepts

- State: $s \in S$; action: $a \in A$; reward: $r(s, a) \in \mathbb{R}$
- Dynamics: $p(s_{t+1} | s_t, a_t)$ for stochastic; $s_{t+1} = f(s_t, a_t)$ for deterministic
- Policy: $\pi(a_t | s_t)$ for stochastic; $a_t = \pi(s_t)$ for deterministic
- Trajectory: $p_\pi(\xi = s_0, a_0, s_1, a_1, \ldots) = p(s_0) \prod_t \pi(a_t | s_t) p(s_{t+1} | s_t, a_t)$
- Return: $R(\xi) = \sum_t \gamma^t r(s_t, a_t) \qquad 0 \leq \gamma < 1$
- Value: $V(s) = \mathbb{E}_{\xi \sim p_\pi}[R | s_0 = s]$

  $Q(s, a) = \mathbb{E}_{\xi \sim p_\pi}[R | s_0 = s, a_0 = a]$

Optimality principle

- **Proposition:** if ξ is a shortest path from s to s_f that goes through s', then a suffix of ξ is a shortest path from s' to s_f
- **Proof:** otherwise, let ξ' be a shorter path from s' to s_f, then take $s \xrightarrow{\xi} s' \xrightarrow{\xi'} s_f$
- It follows that for all $s \neq s_f$ $\quad V(s) = \min_a (1 + V(f(s,a)))$
- The optimal policy is $\quad \pi(s) = \arg\min_a (1 + V(f(s,a)))$

Algorithm 1 Bellman-Ford

$V(s_f) \leftarrow 0$
$V(s) \leftarrow \infty \qquad \forall s \in S \backslash \{s_f\}$
for ℓ from 1 to $|S| - 1$ **do**
  $V(s) \leftarrow \min_{a \in A}\{1 + V(f(s,a))\} \qquad \forall s \in S \backslash \{s_f\}$

Horizon classes

- Finite: $R(\xi) = \sum_{t=0}^{T-1} r(s_t, a_t)$
- Infinite: $R(\xi) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r(s_t, a_t)$
- Discounted: $R(\xi) = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \qquad 0 \leq \gamma < 1$
- Episodic: $R(\xi) = \sum_{t=0}^{T-1} r(s_t, a_t)$ s.t. $s_T = s_f$

Reinforcement Learning — the frontier

- The hard questions in RL:
  - How to perform better exploration?
  - How to model / structure the agent's policy? in particular, its memory
    - Hierarchical RL
  - How to jointly learn multiple tasks?
  - How to learn from more / multiple modalities of data?
    - RL + imitation learning / NLP / vision / program synthesis
  - How to learn in multi-agent environments?
  - How to interface with a human teacher / collaborator?

Upcoming...

logistics

- Join piazza for announcements and forum
- See website for est. schedule, course resources

assignments

- Assignment 1 to be published soon