REINFORCEMENT LEARNING

CP8319/CPS824
Lecture 1

Instructor: Nariman Farsad

* Some of the slides in this deck are adopted from courses offered David Silver, Emma Brunskill, and Sergey Levine.

Today’s Agenda

1. Administrative
   - Please read the course outline in D2L carefully
   - Course website: http://narimanfarsad.com/cps824/index.html

2. Introduction to Reinforcement Learning

Things that you should know

- Linear algebra (MTH 108)
  - matrix multiplication, eigenvector
- Multivariable Calculus (MTH 207)
  - Partial derivative, gradients, Jacobian Matrix
- Probability
  - distribution, random variable, expectation, conditional probability, variance, density
- Basic programming (in Python)
- We will review some of these during first 3 weeks. You should review on your own as well.
  - Good Resource: http://narimanfarsad.com/cps824/background.html

This is a mathematically intense course.
But that's why it's exciting and rewarding!

Differences From Previous Years

- Everything will be online --- lectures, office hours, discussions between students
  - We strongly encourage you to study with other students
  - Technology:
    - D2L discussion boards,
    - **Join the Discord Group**: https://discord.gg/NUvbzUjKcT

- Enrollments increased by ~2x compared to last year
  - About 100 students, ~60 undergrads and ~40 graduate students

Course Evaluation

- Three assignment each 15% (total 45%)
  - have theoretical (math) and practical (programming) questions
  - Are very intensive, start as soon as they are released, or you can’t finish them
  - It is fine to discuss the problems with your classmates, but must write your own solutions
  - CP8319 students get extra questions for each assignment

- Final Project (5% for proposal and 50% for final submission)
  - CPS824 will work in groups of 4 (randomly assigned if not formed by Jan 27)
    - Each group is assigned a TA, who will be their mentor for the projects throughout the semester
  - CP8319 can work individually or in groups of up to 4
    - The instructor will mentor for the projects throughout the semester
  - Final submission evaluation based on TA feedbacks, groupmates feedbacks, code, report, video

Final Project

- Since this is most of your grade, the project you do must be “significant”
  - See this page for more details: http://narimanfarsad.com/cps824/project.html
- Be in communication with your TA mentors and me constantly regarding your project
- We can help you identify a project that is both “significant” and manageable during a semester

Overall Course Structures

- Some of the lectures may be used for breakout sessions for students to work on the project.
- Lectures will be recorded, but I suggest to you attend the lectures (you get to ask questions)
- All of you can succeed if you put in the effort
- We, the class staff, and your fellow classmates, are here to help

Poll: How many of you took ML?

A. I took machine learning with you last semester.

B. I took machine learning but not with you.

C. I have not taken machine learning.

Poll: Why are you taking this course?

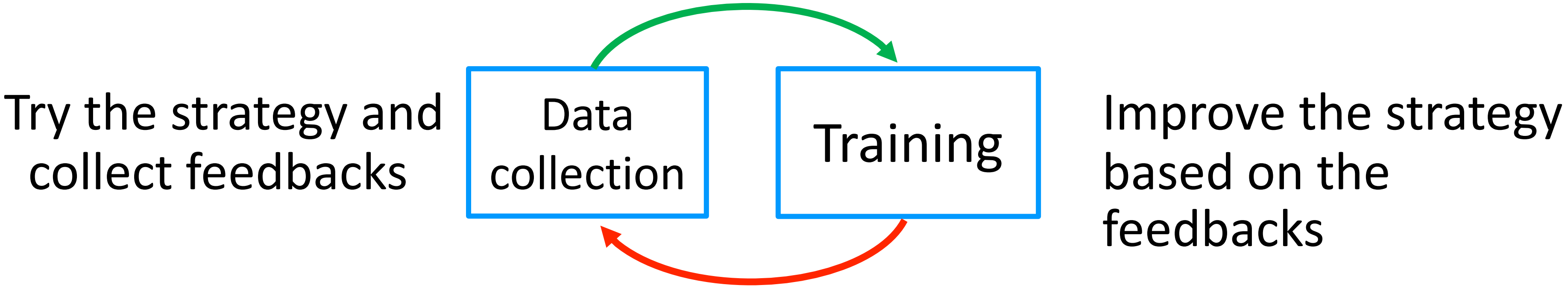
A. To learn to apply reinforcement learning to different problems.

B. To become an expert in reinforcement learning or do research in this field.

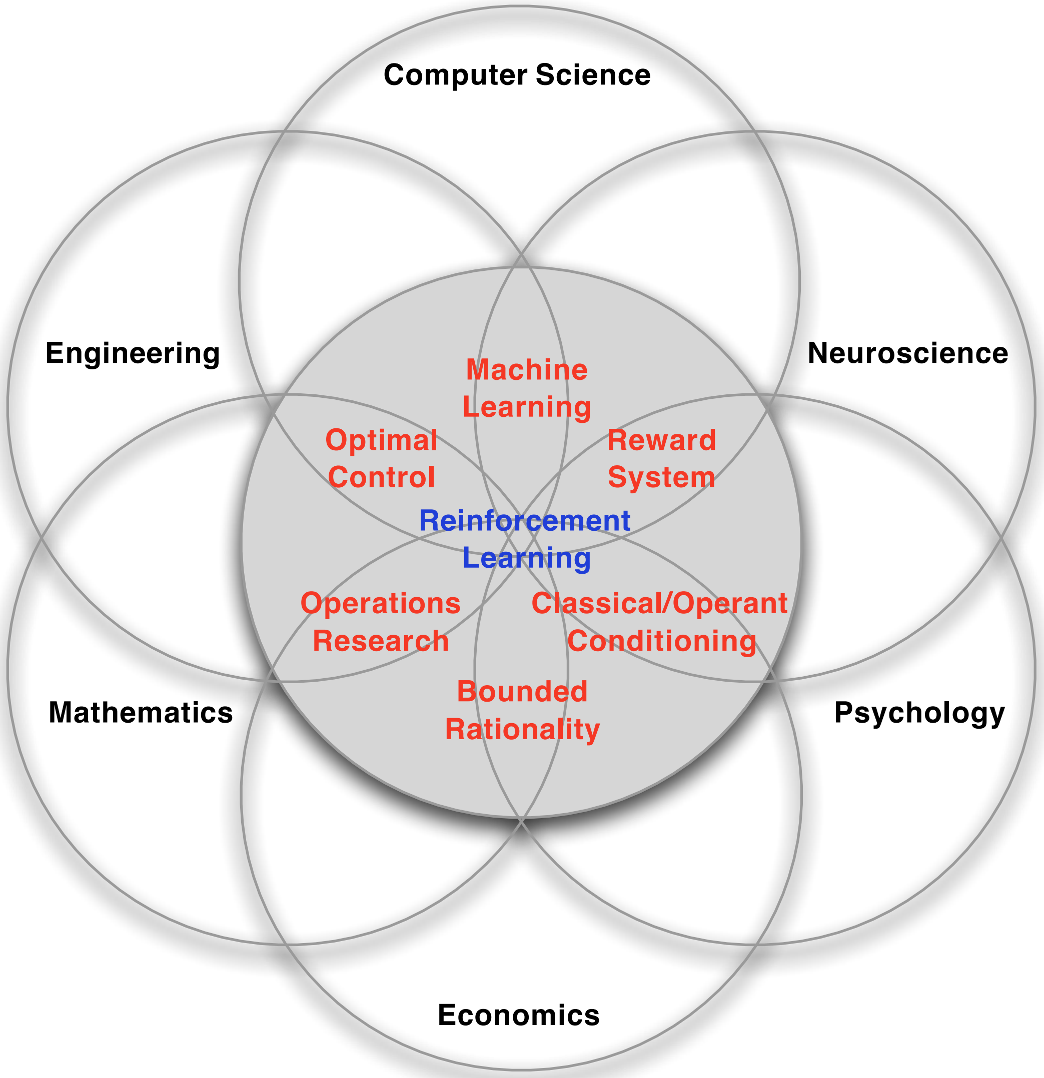
C. I was just curious what is the big deal with reinforcement learning.

Reinforcement Learning

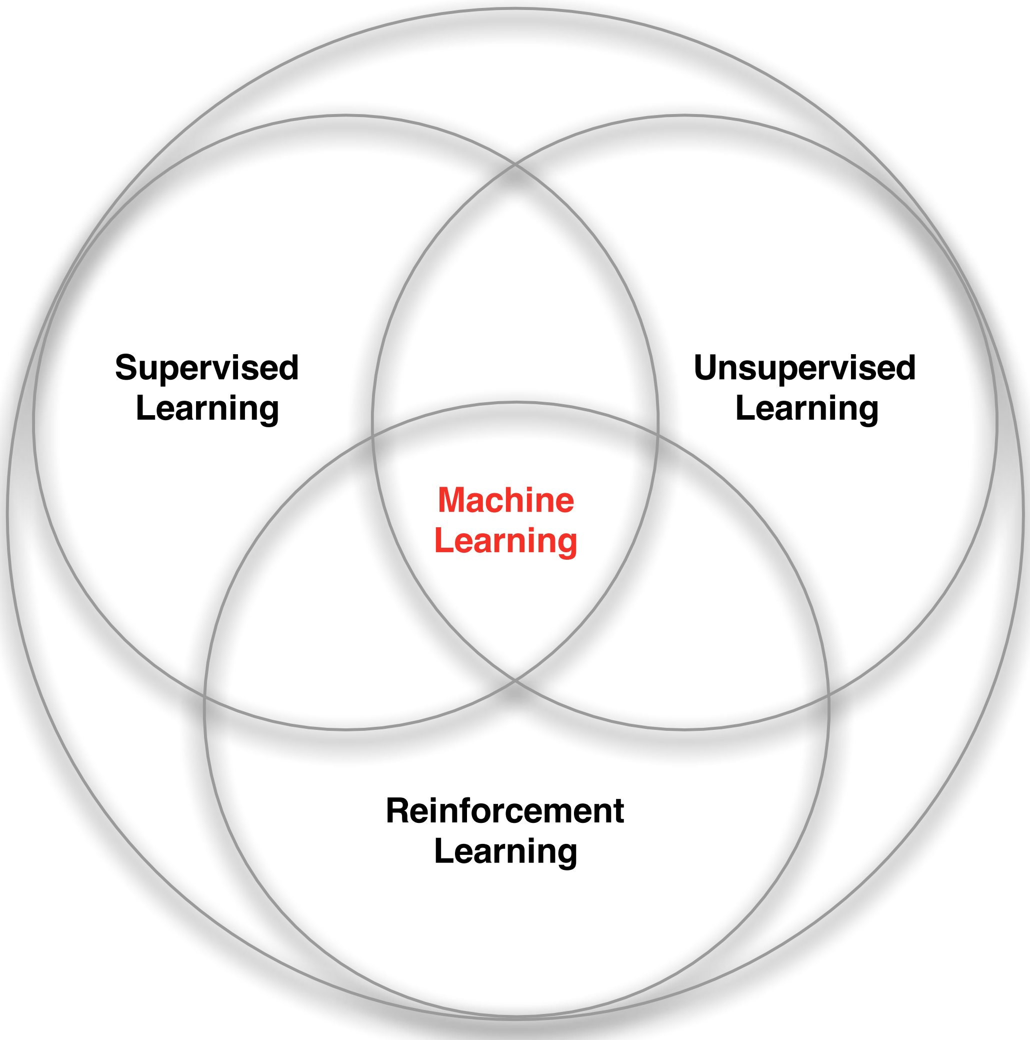
- Learn to make good sequences of decisions under uncertainty (“**science of making decisions**”)



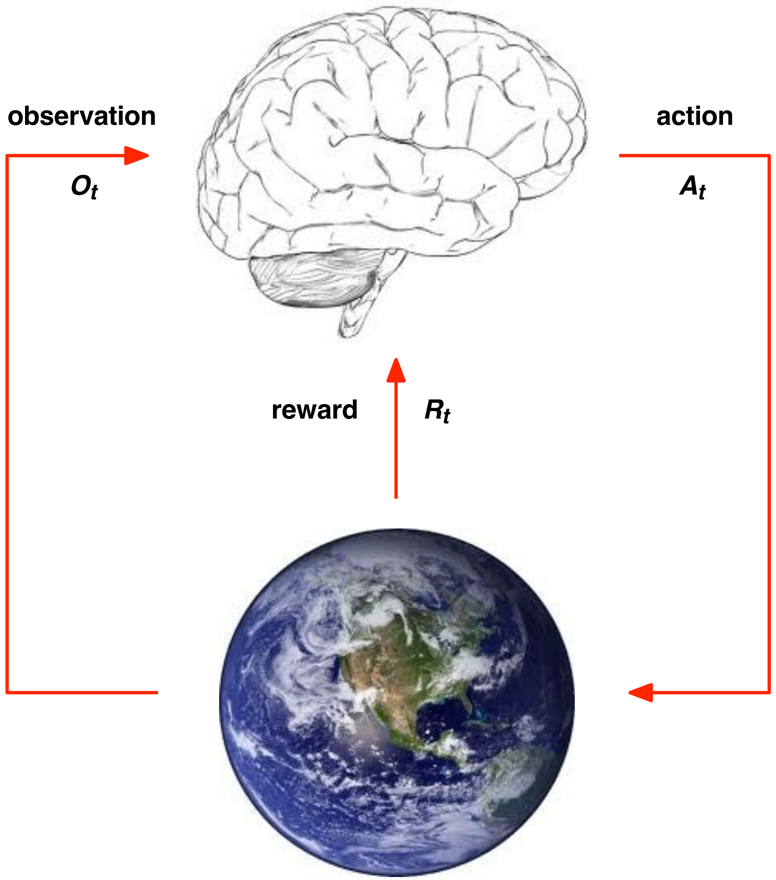
Relation to Other Disciplines



Relation to Machine Learning



RL: The Agent and the Environment



- At each step t the agent:
  - Executes action A_t
  - Receives observation O_t
  - Receives scalar reward R_t
- The environment:
  - Receives action A_t
  - Emits observation O_{t+1}
  - Emits scalar reward R_{t+1}
- t increments at env. step

Reward

- A reward R_t is a scalar feedback signal
- Indicates how well agent is doing at step t
- The agent's job is to maximise cumulative reward
- Reinforcement learning is based on the reward hypothesis

Definition (Reward Hypothesis)
All goals can be described by the maximization of expected cumulative reward

Interesting discussion surrounding this topic:
http://incompleteideas.net/rlai.cs.ualberta.ca/RLAI/rewardhypothesis.html

Examples of Reward

- Fly stunt manoeuvres in a helicopter
  - +ve reward for following desired trajectory
  - −ve reward for crashing
- Defeat the world champion at Backgammon
  - +/−ve reward for winning/losing a gam
- Manage an investment portfolio
  - +ve reward for each $ in bank
- Control a power station
  - +ve reward for producing power
  - −ve reward for exceeding safety thresholds
- Make a humanoid robot walk
  - +ve reward for forward motion
  - −ve reward for falling over
- Play many different Atari games better than humans
  - +/−ve reward for increasing/decreasing score

Sequential Decision Making

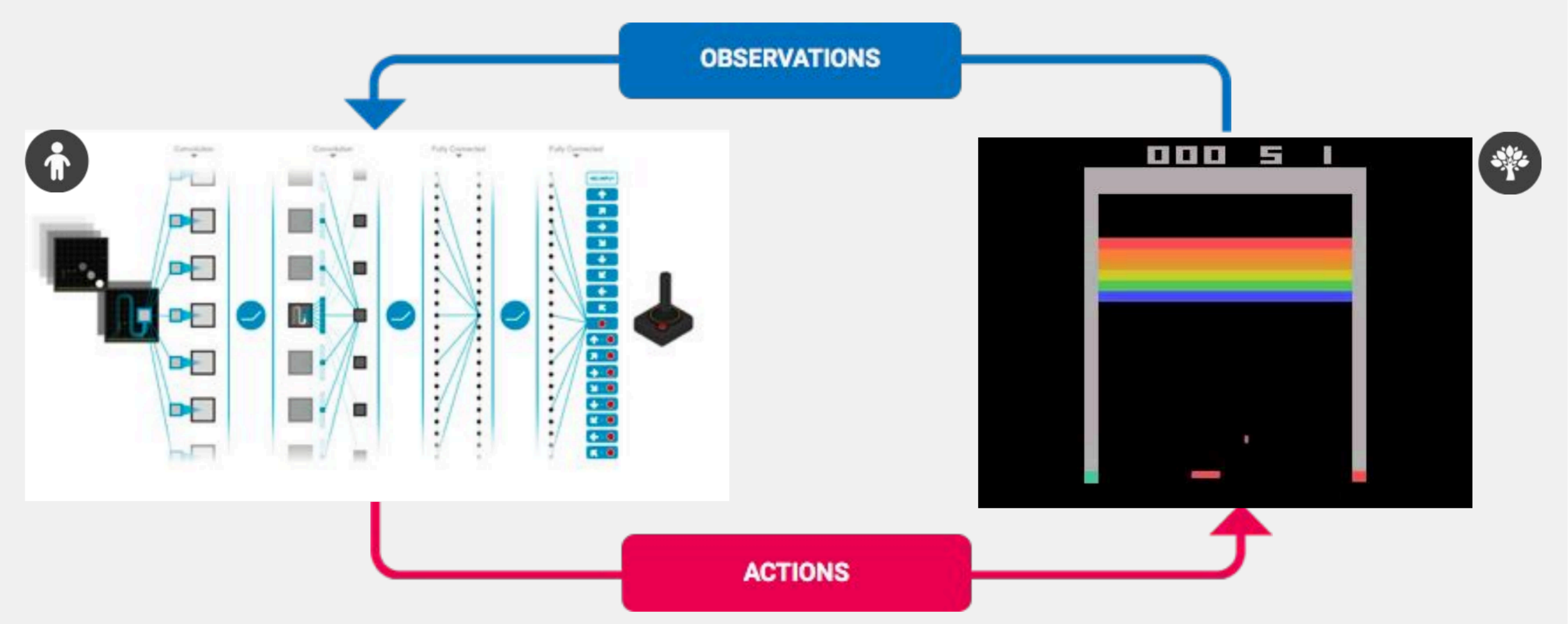
- Goal: *select actions to maximise total future reward*
- Actions may have long term consequences Reward may be delayed
- It may be better to sacrifice immediate reward to gain more long-term reward

- Examples:
  - A financial investment (may take months to mature)
  - Refueling a helicopter (might prevent a crash in several hours)
  - Blocking opponent moves (might help winning chances many moves from now)

RL Examples

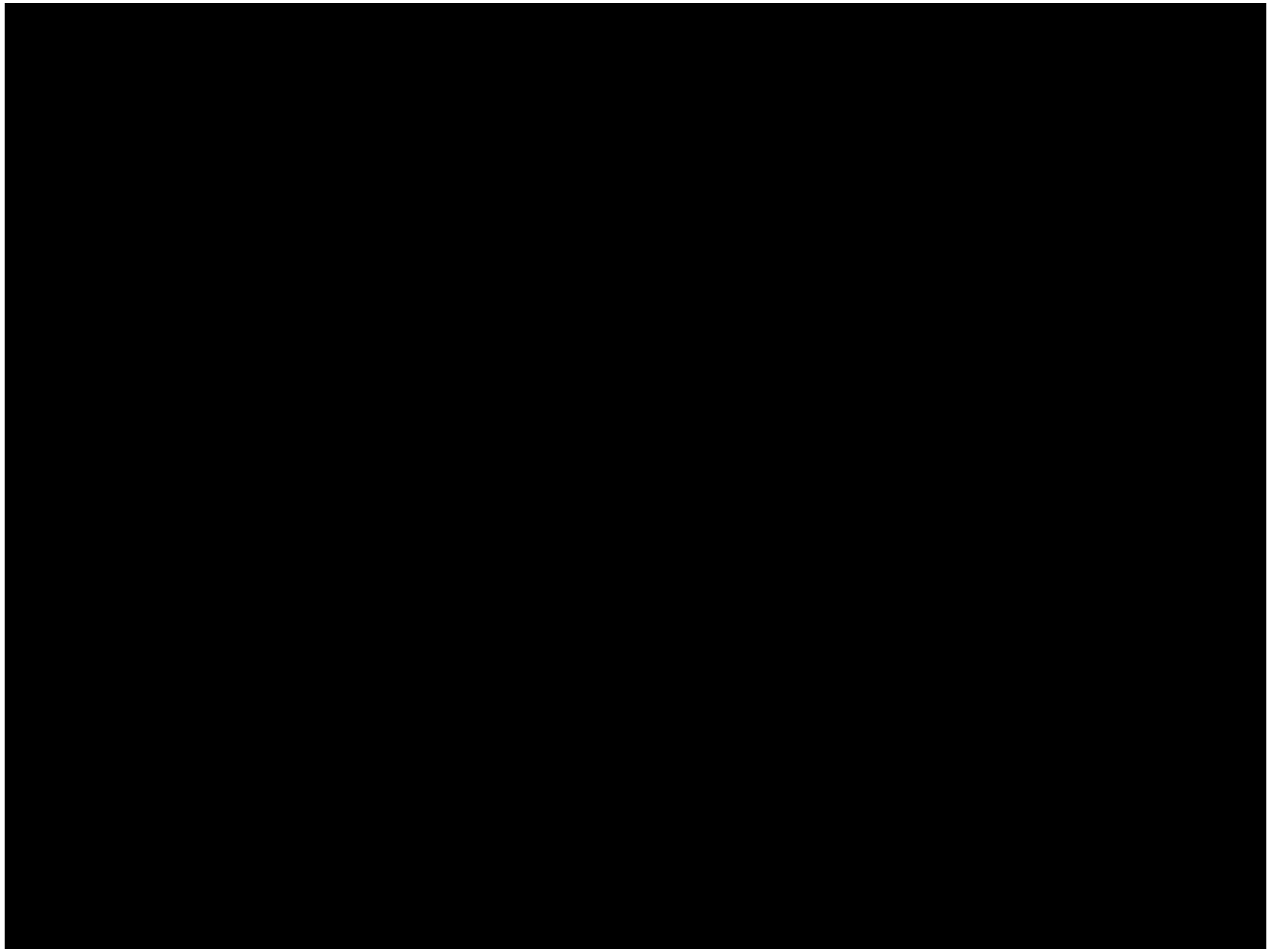
- Fly stunt manoeuvres in a helicopter
- Defeat the world champion at Backgammon
- Manage an investment portfolio
- Control a power station
- Make a humanoid robot walk
- Play many different Atari games better than humans

RL Example: Playing Atari Games



V Mnih *et al.* *Nature* **518**, 529-533 (2015) doi:10.1038/nature14236

RL Example: Playing Atari Games



V Mnih *et al.* *Nature* **518**, 529-533 (2015) doi:10.1038/nature14236

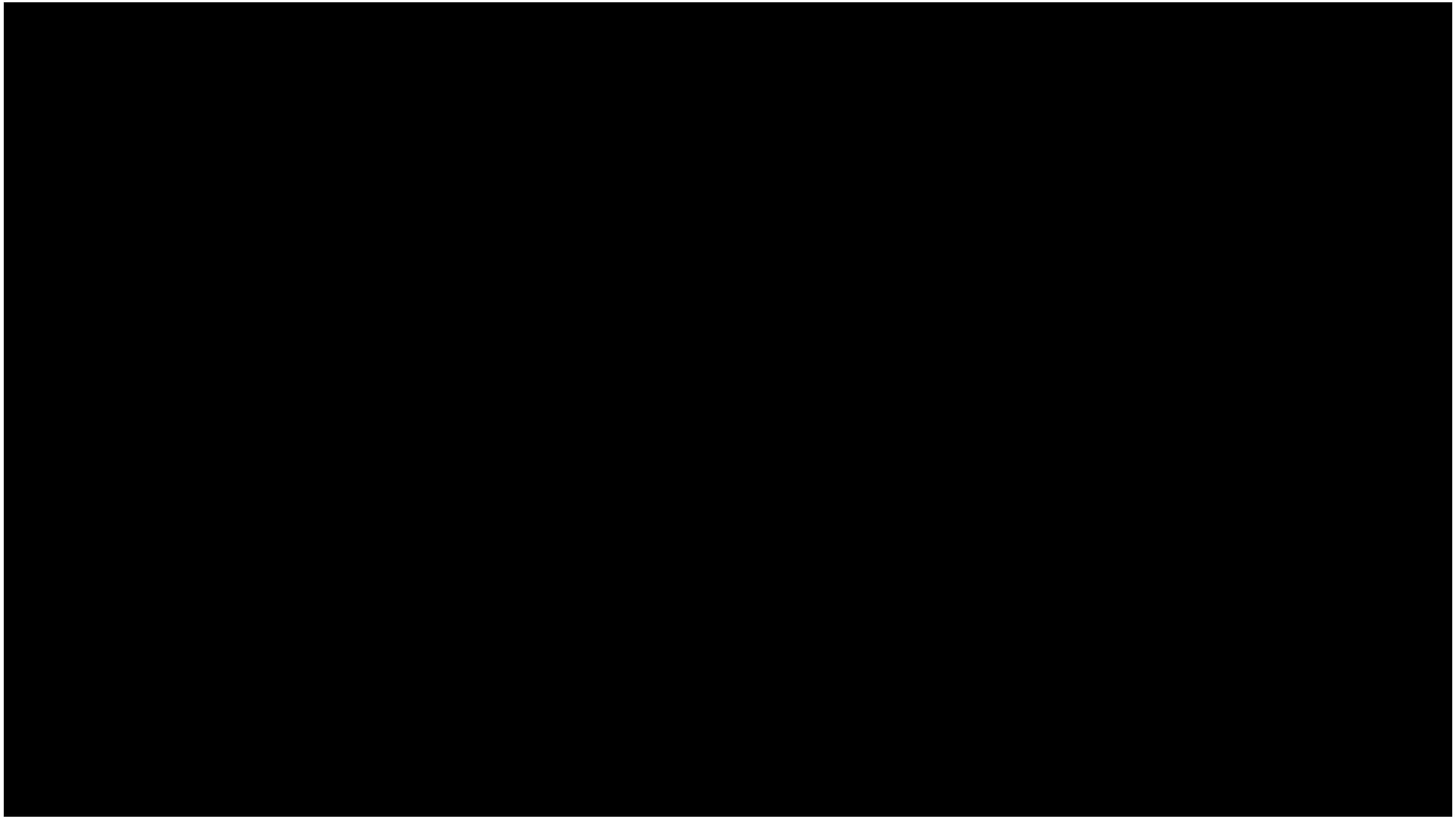
RL Example: Stanford Autonomous Helicopter

- Two controllers

- Can be flown in many ways

- How do we learn to fly?

Stanford Autonomous Helicopter Using RL



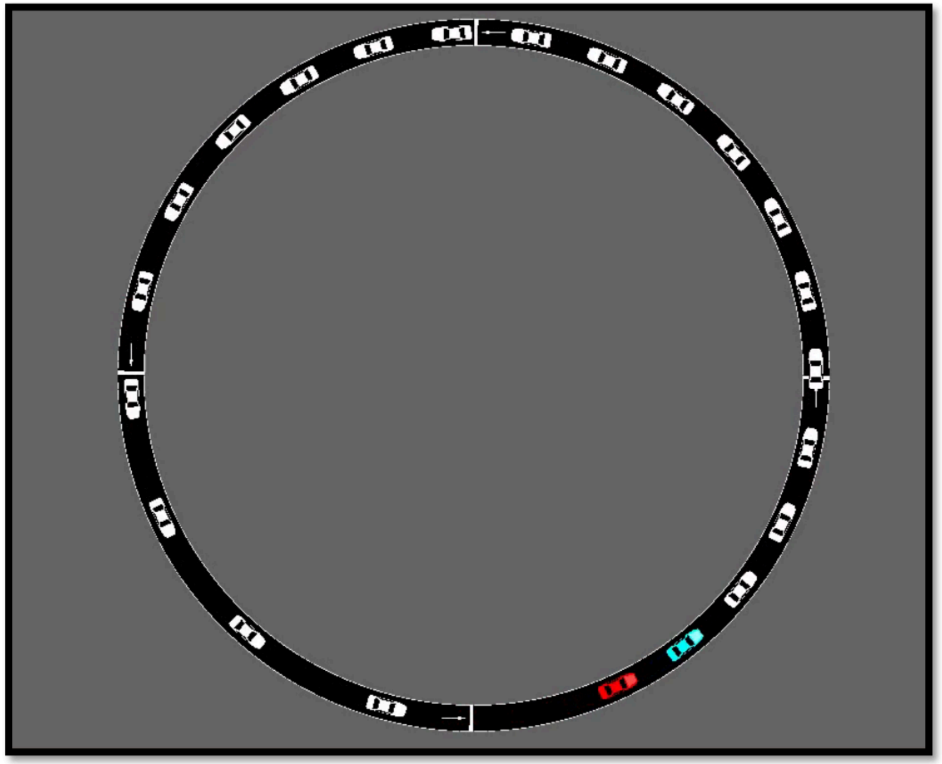
Source: http://heli.stanford.edu/icml2008/

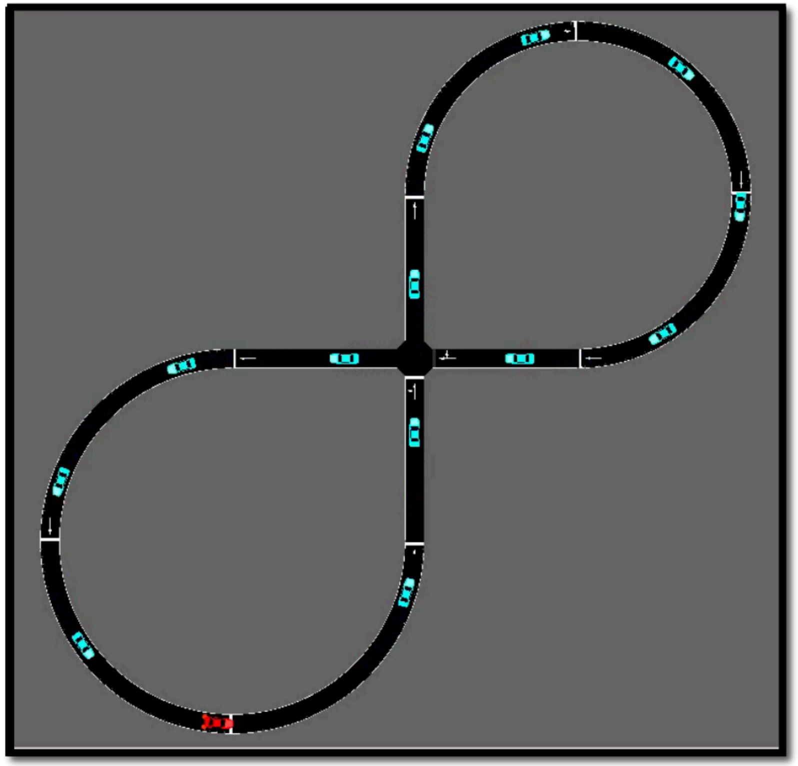
RL Example: Robotics

Source: https://youtu.be/2hGngG64dNM

RL Example: Autonomous Driving

- Come up with a policy based on a leading car (red) such that all cars move at maximum speed, while there are no collisions





Characteristics of RL

What makes reinforcement learning different from other machine learning paradigms?

- There is no supervisor, only a *reward* signal
- Feedback is delayed, not instantaneous
- Time really matters (sequential, non i.i.d data)
- Agent’s actions affect the subsequent data it receives

What We Learn in the Course?

- Markov decision processes & planning
- Model-free policy evaluation
- Model-free control
- Reinforcement learning with function approximation
- Deep RL
- Policy Search
- Exploration
- Advanced Topics (imitation learning, transfer learning, inverse RL, etc.)

See website for more details (syllabus will be updated soon)