Deep Reinforcement Learning, Decision Making, and Control

CS 285
Course logistics
Class Information & Resources

- Course website: [http://rail.eecs.berkeley.edu/deeprlcourse](http://rail.eecs.berkeley.edu/deeprlcourse)
- Piazza: UC Berkeley, CS285
- Gradescope: UC Berkeley, CS285
- Subreddit (for non-enrolled students): [www.reddit.com/r/berkeleydeeprlcourse/](http://www.reddit.com/r/berkeleydeeprlcourse/)
- Office hours: check course website (mine are after class on Wed)

Sergey Levine  
Instructor

Frederik Ebert  
Head GSI

Anusha Nagabandi  
GSI

Avi Singh  
GSI

Kelvin Xu  
GSI
Prerequisites & Enrollment

• All enrolled students must have taken CS189, CS289, CS281A, or an equivalent course at your home institution
  • Please contact Sergey Levine if you haven’t

• If you are not eligible to enroll directly into the class, fill out the enrollment application form: http://rail.eecs.berkeley.edu/deeprlcourse/
  • We will enroll subject to availability based on responses to this form
  • We will not use the official CalCentral wait list!
  • Fill out an application before Friday!

• Lectures are recorded and live streamed (link on course website)
What you should know

• Assignments will require training neural networks with standard automatic differentiation packages (TensorFlow by default)

• Review Section
  • Kelvin Xu will review TensorFlow on Mon in week 3 (Sep 9)
  • You should be able to understand the overview here: https://www.tensorflow.org/guide/low_level_intro
    • If not, make sure to attend Kelvin’s lecture and ask questions!
What we’ll cover

• Material will be similar to previous year:
  http://rail.eecs.berkeley.edu/deeprlcourse-fa18/

1. From supervised learning to decision making
2. Model-free algorithms: Q-learning, policy gradients, actor-critic
3. Advanced model learning and prediction
4. Transfer and multi-task learning, meta-learning
5. Exploration
6. Open problems, research talks, invited lectures
Assignments

1. Homework 1: Imitation learning (control via supervised learning)
2. Homework 2: Policy gradients (“REINFORCE”)
3. Homework 3: Q learning and actor-critic algorithms
4. Homework 4: Model-based reinforcement learning
5. Homework 5: Advanced model-free RL algorithms
6. Final project: Research-level project of your choice (form a group of up to 2-3 students, you’re welcome to start early!)

Grading: 50% homework (10% each), 50% project
5 late days total
Your “Homework” Today

1. Sign up for Piazza (UC Berkeley CS285)
2. Start forming your final project groups, unless you want to work alone, which is fine
3. Review this: https://www.tensorflow.org/guide/low_level_intro
What is reinforcement learning, and why should we care?
How do we build intelligent machines?
Intelligent machines must be able to adapt
Deep learning helps us handle *unstructured environments*
Reinforcement learning provides a formalism for behavior

decisions (actions)

consequences
observations
rewards

Figure 2: An illustration of a normal opening position in Go. A TD
controller has played a near-universal Go championship in the top 10 experts (poker-like opening roll). For example, with an opening roll of 2-4, most players have now switched from the traditional move of 13-5-6-8 to TD controllers’ performance, 13-5-4-2. TD controller analysis is given in Table 2.

Schulman et al. ’14 & ‘15
Mnih et al. ‘13
Levine*, Finn*, et al. ‘16

Mnih et al. ‘13
Levine*, Finn*, et al. ‘16
What is deep RL, and why should we care?

- **Standard computer vision**
  - Features (e.g. HOG)
  - Mid-level features (e.g. DPM)
  - Classifier (e.g. SVM)
  - *Felzenszwalb '08*

- **Deep learning**
  - End-to-end training

- **Standard reinforcement learning**
  - Features
  - More features
  - Linear policy or value func.

- **Deep reinforcement learning**
  - End-to-end training
  - Action
What does end-to-end learning mean for sequential decision making?
Perception

Action (run away)

Action
sensorimotor loop

Action
(run away)
Example: robotics

robotic control pipeline

- observations
- state estimation (e.g. vision)
- modeling & prediction
- planning
- low-level control
- controls
tiny, highly specialized “visual cortex”

tiny, highly specialized “motor cortex”

sensorimotor loop
Deep models are what allow reinforcement learning algorithms to solve complex problems end to end!

The reinforcement learning problem is the AI problem!
Complex physical tasks...
Unexpected solutions...

Mnih, et al. 2015
In the real world...

Kalashnikov et al. ‘18
In the real world...
Not just games and robots!

Cathy Wu
Why should we study this now?

1. Advances in deep learning
2. Advances in reinforcement learning
3. Advances in computational capability
Why should we study this now?


Tesauro, 1995
Why should we study this now?

Atari games:
Q-learning:

Policy gradients:

Real-world robots:
Guided policy search:

Q-learning:

Beating Go champions:
Supervised learning + policy gradients + value functions + Monte Carlo tree search:
What other problems do we need to solve to enable real-world sequential decision making?
Beyond learning from reward

- Basic reinforcement learning deals with maximizing rewards
- This is not the only problem that matters for sequential decision making!
- We will cover more advanced topics
  - Learning reward functions from example (inverse reinforcement learning)
  - Transferring knowledge between domains (transfer learning, meta-learning)
  - Learning to predict and using prediction to act
Where do rewards come from?

reinforcement learning agent

As human agents, we are accustomed to operating with rewards that are so sparse that we only experience them once or twice in a lifetime, if at all.
Are there other forms of supervision?

- Learning from demonstrations
  - Directly copying observed behavior
  - Inferring rewards from observed behavior (inverse reinforcement learning)
- Learning from observing the world
  - Learning to predict
  - Unsupervised learning
- Learning from other tasks
  - Transfer learning
  - Meta-learning: learning to learn
More than imitation: inferring intentions

Warneken & Tomasello
Inverse RL examples

Finn et al. 2016
“the idea that we predict the consequences of our motor commands has emerged as an important theoretical concept in all aspects of sensorimotor control”
Prediction for real-world control
Using tools with predictive models

Xie et al. 2019
Playing games with predictive models

But sometimes there are issues...

Kaiser et al. 2019
How do we build intelligent machines?
How do we build intelligent machines?

• Imagine you have to build an intelligent machine, where do you start?
Learning as the basis of intelligence

• Some things we can all do (e.g. walking)
• Some things we can only learn (e.g. driving a car)
• We can learn a huge variety of things, including very difficult things
• Therefore our learning mechanism(s) are likely powerful enough to do everything we associate with intelligence
  • But it may still be very convenient to “hard-code” a few really important bits
A single algorithm?

- An algorithm for each “module”?
- Or a single flexible algorithm?

[BrainPort; Martinez et al; Roe et al.]

adapted from A. Ng
What must that single algorithm do?

• Interpret rich sensory inputs

• Choose complex actions
Why deep reinforcement learning?

• Deep = can process complex sensory input
  ▪ ...and also compute really complex functions

• Reinforcement learning = can choose complex actions
Some evidence in favor of deep learning

Unsupervised learning models of primary cortical receptive fields and receptive field plasticity

Andrew Saxe, Maneesh Bhand, Ritvik Mudur, Bipin Suresh, Andrew Y. Ng
Department of Computer Science
Stanford University
{asaxe, mbhand, rmudur, bipins, ang}@cs.stanford.edu
Some evidence for reinforcement learning

• Percepts that anticipate reward become associated with similar firing patterns as the reward itself

• Basal ganglia appears to be related to reward system

• Model-free RL-like adaptation is often a good fit for experimental data of animal adaptation
  • But not always...
What can deep learning & RL do well now?

• Acquire high degree of proficiency in domains governed by simple, known rules
• Learn simple skills with raw sensory inputs, given enough experience
• Learn from imitating enough human-provided expert behavior
What has proven challenging so far?

• Humans can learn incredibly quickly
  • Deep RL methods are usually slow
• Humans can reuse past knowledge
  • Transfer learning in deep RL is an open problem
• Not clear what the reward function should be
• Not clear what the role of prediction should be
Instead of trying to produce a program to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain.

- Alan Turing