CS 294 Deep RL — UCB, Spring 2017 — S. Levine —- Scribe: William Guss
Lecture Notes

Course Info.

rll.berkeley.edu piazza berkeley cs294-112

What you should know. Assignments will require trianing neural networks. You should know a standard automatic differentiation tool.

What we'll cover. (Full syllabus on course website.)

- From supervised learning to decision making.
- Basic reinforcmeent learning: Q-learniong and policy gradients.— Should help get you the basics!
- Advanced model learning and prediction, distillation, reward learning.
- Advanced deep RL: trust region policy gradients, actor critic methods, exploration
- Open problem, research talks, invited lectures.

Assignments

- Homework 1: Imitation learning (control via supervised learning)
- Homework 2: Basic (shallow) RL
- Homework 3: Deep Q-Learning
- Homework 4: Deep policy gradients.
- Final project: Research-level project of your choice (form a group of up to 2-3 students, you're welcome to start early!)

Grading for this class is: 40% homework (10% each), 50% project, 10% participation. *Pretty nice!*.

How do we build intelligent machines? — Sergey Levine

- Imagine you have to build an intelligent machine, where do you start? You might start by looking at a persons brain and break it into different functions. Then you would some how integrate all of these modules, ie. translate these modules into seperate pieces of code! *Sergey:* This is an 'early' perspective on intelligence. We'd like something more, this is just too complicated.
- **Instead:** Learning as the basis of intelligence.
  - Some things we can all do (eg. walking), but most intelligent behaviours are emergent as a result of our 'nurture.'
  - Some things we can only learn (eg. driving) a car.
  - We can learn a huge variety of things including very difficult things.
  - Therefore our learning mechanisms are likely powerful enough to do everything we associate with intelligence. (We may need to bootstrap a few things...)
- **A single algorithm?**
  - An algorithm for each 'module'?
  - What must this single algorithm do? It needs to be able to interpret rich sensory inputs (I really don't like this idea—in general, there's got to be more!)
  - Perception is really important here.
- **Why deep reinforcement learning?**
  - Deep = can process complex sensory input (and complicated functions in general)
  - Reinforcement learning = can choose complex actions.