introduction to deep RL

- you know what RL is: it provides frameworks for solving sequential decision making type problems.
- you know what deep learning is.
- deep RL is using deep neural nets for solving decision making problems using the conceptualization developed by RL.
- Sergey motivates deep reinforcement learning from a "learn what matters point of view". People often build robotic systems as a pipeline. Each step is like perception, state estimation, planning, control etc. This is a pipeline approach where each step is building a model (abstraction) of the world that is believed to matter for solving a particular task. The downside here is we need to capture the right details (factors of variation) that matter. This is all manually done. Instead applying deep learning to this decision making task can learn the right features that matter → that's the motivation. The representation/abstraction you learn is optimally learnt for that particular task (hopefully).
- The problem of reinforcement learning is to maximize expected rewards over time. This is the same as the problem statement in optimal control, except that the optimal control folks formulate it as minimizing cost instead of maximizing reward.
- This approach is not new to robotics. 10 yrs ago the best approaches to vision did manual feature extraction like edges, etc. and then deep learning was applied and it produced much better results.
- deep RL: raw sensory percepts -> control outputs
- in theory, we can re-formulate a supervised learning problem as a RL problem. how? we can reformulate the inputs as observations, outputs as actions and loss function as rewards. this may not always be a good idea because RL makes fewer assumptions and is generally harder to solve, so you can reformulate this way but why would you if the problem is clear and we don't need this way of formulating it that makes it harder to solve. eg. image recognition. the decisions are not sequential, future actions and current actions are not related. score is based on individual actions, you get the idea.
- the reinforcement learning problem is THE AI problem. you can often formulate other problems from a RL perspective although again, although that's not always the best way to go about things.
- basic RL (like we covered in cs 229) deals with maximizing rewards. this doesn't always work for sequential decision making -> which is actually the thing we're concerned with here. apart from RL needed for sequential decision making, we will also look at other things that help us with the larger goal of sequential decision making like learning reward functions from observations (inverse reinforcement learning -> figuring out rewards from demonstrations(actions)) since this is often hard to correctly specify, directly copying observed behavior, tranferring skills between domains, learning to predict which means sometimes we don't experience the reward function enough to learn just from it but we'll learn how the world works and use this to our advantage to figure out what actions might result in what changes in the world to see what would lead us to our goal. Hard to express rewards and not seeing the reward enough times turns up everywhere and figuring out how to work around these can be tremendously valuable.
- again, i want to reinforce how exciting prediciton looks from the perspective of the classical exploration/exploitation problem. if we can predict better, we can pay fewer costs of bad exploration. we can either learn to predict (by learning a model) or we can already have a near perfect model in some tasks. if we already have a model, we can do more complicated tasks (model based RL) than if we had to learn the model used for predictions of what will happen if we take certain actions.

- apart from these practical things that seem of huge promise, deep RL is also exciting because it might help us understand more about intelligence, humans and how to build general intelligence machines.
- learning as the basis of intelligence:
  - we can learn to do a huge variety of pretty complicated things → driving, cooking, playing tennis really well etc.
  - our learning mechanisms might be powerful enough to do everything we think of as intelligence
  - therefore, it might just be that intelligence is the ability to learn. so if we can teach a robot to do really good RL, that might just be intelligence!
- ppl have learned to perceive the world with their tongue with a electrode connected to it! evidence for a singular general learning algorithm that has learned different tasks? what must that single algorithm do?
  - perception: sight, sound, taste, touch, smell
  - perform actions in the world
- some evidence in favor of deep RL's relationship to features learned by some parts of the brain: some features are similar! this doesn't mean the brain does what deep RL systems do → it just means the features are the same. this could just be because the input environment is the same for the brain and the deep RL system.
- what do the modern deep RL systems still find challenging?
  - they take a very long time to learn compared to humans. this might be because they don't re-use past knowledge of how things work?
    - transfer learning in deep rl is an open problem
  - sometimes not clear what the reward function should be
  - sometimes not clear how to fit together prediction with rl (clearly humans have some model of the world, they can do predictions, they don't do model free RL)
- the word deep in a context suggests that a neural network is used for some function in that context. in deep rl the functions we could use a neural net for are:
  - the policy
  - the value function (measures goodness of state or state-action pairs)
  - dynamics model → the prediction model, how does the world work.
- RL is different from supervised learning in that:
  - you don't have direct access to the loss function. you kinda have a reward after a while and you have to learn what is good what is bad over time
  - supervised learning is kind of a i.i.d world. RL → your inputs are dependent on past actions.
-