R9/Sanit Gupta/160100010

August 6, 2020

In this paper, the authors address the problem of Inverse Reinforcement Learning (IRL). IRL is the problem setting where one must infer the reward function by just observing the optimal behaviour. This has clear applications. In any environment, the reward function may or may not be obvious. Instead of making assumptions, that might be unwarranted, and building a reward function on one's own, one can observe an agent that is doing well in the environment and infer a reward function using IRL.

In this paper, the authors introduce three different for three different settings. The biggest problem they observe across the different settings is that of degeneracy. There isn't a one-to-one mapping between optimal behaviour and reward functions. There's a ton of reward functions that share the same optimal behaviour, therefore one can't guarantee the correctness of the reward function inferred by observing optimal behaviour. To deal with this, the authors develop heuristics for the three settings aiming to pick the reward function which maximally separates the observed policy from other policies.

In their experiments, their algorithms worked extremely well, at least for problems that were not very large.

The three settings were:

- The first case is the simplest one with a finite state MDP. In this case, both the transition matrix and the optimal policy are known. They begin by characterizing the set of reward functions for which the policy would be optimal. This characterization is a set of linear constraints and linear programming can be used to find a feasible point. They design a criteria which gives high values for reward functions in which sub-optimal policies are much worse than the optimal one. They also add a regularization term which penalizes reward functions which have high rewards. The reward function that maximises this criteria is the one they choose.
- The second case is simply the extension of the previous case to infinite state spaces. Now, a linear approximation is used for the reward function. Not much else is different from the previous case.
- In the third case, one doesn't know the exact optimal policy. Instead, one has only observed some trajectories of the optimal policy. This is clearly

the more realistic case. They assume that they can simulate trajectories of any policy of their choice. Is this assumption reasonable?

Some questions I had:

- What would one do if one is trying to learn the reward function from a policy that is not necessarily optimal (e.g. if it is epsilon optimal instead)? I think it might be unreasonable to believe that an actual agent one observes in the environment is behaving optimally
- How would one integrate one's own beliefs about the reward function into this?