README.md

MDPs and Bellman Equations

Learning Goals

- Understand the Agent-Environment interface
- Understand what MDPs (Markov Decision Processes) are and how to interpret transition diagrams
- Understand Value Functions, Action-Value Functions, and Policy Functions
- Understand the Bellman Equations and Bellman Optimality Equations for value functions and action-value functions

Summary

- Agent & Environment Interface: At each step `t` the agent receives a state `S_t`, performs an action `A_t` and receives a reward `R_{t+1}`. The action is chosen according to a policy function `pi`.
- The total return `G_t` is the sum of all rewards starting from time t. Future rewards are discounted at a discount rate `gamma^k`.
- Markov property: The environment's response at time `t+1` depends only on the state and action representations at time `t`. The future is independent of the past given the present. Even if an environment doesn't fully satisfy the Markov property we still treat it as if it is and try to construct the state representation to be approximately Markov.
- Markov Decision Process (MDP): Defined by a state set S, action set A and one-step dynamics `p(s',r | s,a)`. If we have complete knowledge of the environment we know the transition dynamic. In practice, we often don't know the full MDP (but we know that it's some MDP).
- The Value Function `v(s)` estimates how "good" it is for an agent to be in a particular state. More formally, it's the expected return `G_t` given that the agent is in state `s`. `v(s) = Ex[G_t | S_t = s]`. Note that the value function is specific to a given policy `pi`.
- Action Value function: q(s, a) estimates how "good" it is for an agent to be in states and take action a. Similar to the value function, but also considers the action.
- The Bellman equation expresses the relationship between the value of a state and the values of its successor states. It can be expressed using a "backup" diagram. Bellman equations exist for both the value function and the action value function.
- Value functions define an ordering over policies. A policy `p1` is better than `p2` if `v_p1(s) >= v_p2(s)` for all states s. For MDPs, there exist one or more optimal policies that are better than or equal to all other policies.
- The optimal state value function `v*(s)` is the value function for the optimal policy. Same for `q*(s, a)`. The Bellman Optimality Equation defines how the optimal value of a state is related to the optimal value of successor states. It has a "max" instead of an average.

Lectures & Readings

Required:

- Reinforcement Learning: An Introduction - Chapter 3: Finite Markov Decision Processes
- David Silver's RL Course Lecture 2 - Markov Decision Processes (video, slides)

Exercises

This chapter is mostly theory so there are no exercises.