Reinforcement Learning

Keith A. Lewis

November 21, 2019

Abstract

Notes based on Sutton and Barto's book.

Reinforcement Learning

Maximizing gains for an agent interacting with a model using goal directed learning. There is always a model but there is no canonical measure of gain.

Markov Decision Process

A MDP is defined by states, S, actions, A, rewards, $R \subseteq \mathbf{R}$, and transition probabilities, $p(s', r'|s, a) = P(S_{t+1} = s', R_{t+1} = r' \mid S_t = s, A_t = a)$, the probability of moving to state s' and receive reward r' given the agent is in state s and takes action a at time t, $s \overset{a/r}{\to} s'$.

Some models specify $A_s \subseteq A$, for $s \in S$, the set of possible actions when in state s.

At time t the agent chooses an action a. This results in a new state, s', and reward, r', at time $t + 1$ according to the transition probabilities.

A *policy*, $\pi(a|s)$, specifies the probability of taking action a given the agent is in state s. This results in the sequence of random variables S_{t+k+1}, R_{t+k+1}, $k \geq 0$, given $S_t = s$.

Reinforcement learning is the study of how to find the optimal policy for a given definition of optimal.

A *gain* (or *loss*) function is any function of future rewards,

$$G_t = g_t(R_{t+1}, R_{t+2}, \ldots).$$

Common choices are are average rewards

$$G_t = (1/k) \sum_{j=1}^{k} R_{t+j+1}$$

and exponential decay

$$G_t = \sum_{k \geq 0} \gamma^k R_{t+k+1},$$

where $0 < \gamma < 1$ is the *discount factor*.

The *state-value function* for policy π is $V_\pi(s) = E[G_t \mid S_t = s]$. (Note that, by the Markov property, it does not depend on t.) We want to find $V^*(s) = \max_\pi V_\pi(s)$. Note

$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s',r'} p(s', r'|s, a)[r' + \gamma V_\pi(s')]$$

for exponential decay and

$$V^*(s) = \max_{a \in A_s} \sum_{s',r'} p(s', r'|s, a)[r + \gamma V^*(s')],$$

is called the *Bellman optimality equation*.

The *action-value function* for π is $Q_\pi(s, a) = E[G_t \mid S_t = s, A_t = a]$. We want to find $Q^*(s, a) = \max_\pi Q_\pi(s, a)$. Note

$$Q^*(s, a) = E[R_{t+1} + \gamma V^*(S_{t+1})|S_t = s, A_t = a].$$

gives the optimal value function.

Bandits

An n-armed bandit is a MDP with one state and n actions. The general idea behind a solution is to *explore* the n available actions and *exploit* the most promising. In this case the action-value function does not depend on the state. If we knew the reward distributions for each action then the optimal strategy would be to always select the action with the largest expected value.

The ϵ-*greedy* strategy selects the action maximizing the current action-value function with probability $1 - \epsilon$ and a random action with probability ϵ. The action-value function is updated based on the observed reward.

Monte Carlo Methods

The *first visit* Monte Carlo prediction approximates the state-value function for a given policy. Choose an initial state-value function $V(s)$. Generate a run using policy π. For each state in the run, $V(s)$ is the average of the returns following s.

Temporal Distance Learning

p(s',r'|s,a) s – a/r' –> s'

p(a|s) only works because Markov

MDP is MAB conditioned on state