★ Introduction to RL

— Reference: FML Chapter 14

Sutton and Barto "Reinforcement learning"

Outline:

1. Intro
2. Model-based and Model-free RL
3. Temporal difference (TD) methods
4. Function approximation for Value function
5. Actor-critic methods
6. (TBD)

(1) Different learning frameworks

Supervised : learning from a training set of labelled example

Unsupervised : find hidden structure in data, estimate density function

Reinforcement : learns from interaction, not from examples
goal is to max reward, not to find hidden structure

(2) Learning from interaction

① learn what to do
learn actions to max a numerical reward

② The agent is not told what to do, but it must discover the best behavior

③ The actions that it takes affect future outcome

(3) Exploration and exploitation dilemma

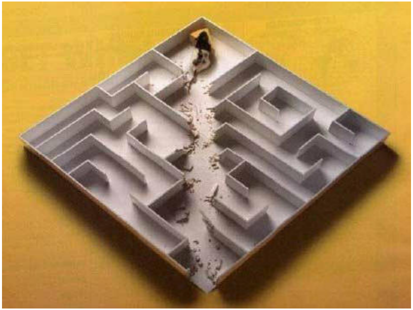
In RL a goal-seeking agent must simultaneously
- exploit current knowledge
- explore new actions

(4) Abstraction :
RL offers an abstraction to the problem of goal-directed learning from interaction.

Learning from interaction

- Reinforcement learning involve learning what to do
- It maps solutions to actions as to maximize a numerical reward
- The agent is not told what to do but it must discover the best behaviour
- The actions that it takes affect future outcomes

Learning from interaction in practise



- Reinforcement learning in practise gives only an approximation to a true solution
- Real problem might be continuous and high dimensional

Exploration and exploitation dilemma

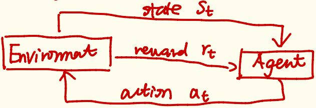
In reinforcement learning we have a goal-seeking agent that must simultaneously:

- **exploit** current knowledge
- **explore** new actions

The agent must try a variety of actions and progressively favour those that appear to be best.

It proposes that the sensory, memory and control apparatus and the objective can be reduced to states, actions and rewards passing back and forth between the agent and the environment.

The agent-environment interface



Reward hypothesis:

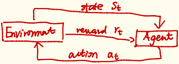
maximise the expected value of the cumulative reward

RL: abstraction

state space $S=\{s^1,\cdots,s^{|S|}\}$

action space $A=\{a^1,\cdots,a^{|A|}\}$

reward space $\mathbb{R}$.



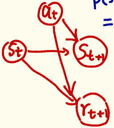
history $h_t=\{s_0,a_0,r_1,s_1,a_1,r_2,\cdots,s_{t-1},a_{t-1},r_t,s_t,a_t\}$

state contains all information about the environment to the agent.

transition model: $\Pr(s_{t+1}=s', r_{t+1}=r \mid h_t) \overset{\text{Markov property}}{=} \Pr(s_{t+1}=s', r_{t+1}=r \mid s_t=s, a_t=a)$

policy: $\Pr(a_{t+1} \mid h_t, s_{t+1})$

Markov property: s_{t+1} only depends on s_t and a_t



$p(s',r|s,a)$

$= \Pr(s_{t+1}=s', r_{t+1}=r \mid h_t) = \Pr(s_{t+1}=s', r_{t+1}=r \mid s_t, a_t)$

$r(s,a)$

$= \mathbb{E}[r_{t+1} \mid s_t=s, a_t=a] = \sum_{r,s'} r\, p(s',r|s,a)$

① state-transition probability $p(s'|s,a)=\sum_r p(s',r|s,a)$

② expected reward $r(s,a,s')=\mathbb{E}[r_{t+1} \mid s_t=s, a_t=a, s_{t+1}=s']$

$r(s,a,s')=\sum_r r\, p(r|s,a,s') = \sum_r r \frac{p(s',r|s,a)}{p(s'|s,a)}$

policy: $\Pr(a_{t+1}|s_{t+1}) = \Pr(a_t|s_t) = \pi(a|s)$

state-value function for policy π

$$V^\pi(s) = \mathbb{E}_\pi[R_t \mid s_t=s]$$

action-value function for policy π

$$Q^\pi(s,a) = \mathbb{E}_\pi[R_t \mid s_t=s, a_t=a]$$

Return (accumulated future reward) $R_t=\sum_{k=0}^{T-t-1}\gamma^k r_{t+k+1}$

Bellman Equation (under Markov Property)

$$V^\pi(s) = \mathbb{E}_\pi[R_t \mid S_t = s] = \mathbb{E}_\pi[r_{t+1} + \gamma R_{t+1} \mid S_t = s]$$

$$= \underbrace{\mathbb{E}_\pi[r_{t+1} \mid S_t = s]}_{(A)} + \gamma \underbrace{\mathbb{E}_\pi[R_{t+1} \mid S_t = s]}_{(B)}$$

$$(A) = \sum_r r\, p(r|s) = \sum_r r \sum_{s',a} p(r, a, s' \mid s)$$

$$= \sum_r r \sum_{s',a} p(s', r \mid s, a)\, \pi(a|s)$$

$$= \sum_r r \sum_a \pi(a|s) \sum_{s'} p(s', r \mid s, a)$$

$$(B) = \mathbb{E}_\pi[R_{t+1} \mid S_t = s]$$

$$= \sum_{s'} \mathbb{E}_\pi[R_{t+1} \mid S_{t+1} = s']\, P_r(S_{t+1} = s' \mid S_t = s)$$

$$= \sum_{s'} \sum_r V^\pi(s')\, P_r(S_{t+1} = s', r_{t+1} = r \mid S_t = s)$$

$$= \sum_{s'} \sum_r V^\pi(s') \sum_a P_r(S_{t+1} = s', r_{t+1} = r \mid S_t = s, a_t = a)\, \pi(a|s)$$

$$= \sum_r \sum_a \pi(a|s) \sum_{s'} p(s', r \mid s, a)$$

$$\Rightarrow V^\pi(s) = \sum_a \pi(a|s) \sum_r \sum_{s'} p(s', r \mid s, a)\left[r + \gamma V^\pi(s')\right]$$

Optimal Value function:

$$V^*(s) = \max_\pi V^\pi(s)$$

$$Q^*(s,a) = \max_\pi Q^\pi(s,a)$$

$$Q^*(s,a) = \mathbb{E}[r_{t+1} | s_t = s, a_t = a] + \gamma \sum_{s',r} V^*(s') P(s',r|s,a).$$

$$= \max \mathbb{E}[R_t \mid s_t = s, a_t = a]$$

$$= \max \mathbb{E}[r_{t+1} + \gamma R_{t+1} \mid s_t = s, a_t = a]$$

$$\max_\pi = \mathbb{E}[r_{t+1} | s_t = s, a_t = a] + \gamma \mathbb{E}[R_{t+1} | s_t = s, a_t = a]$$

$$\max_\pi = \mathbb{E}[r_{t+1} + \gamma R_{t+1} \mid s_t = s, a_t = a]$$

$$\max_\pi = \mathbb{E}[r_{t+1} | s_t = s, a_t = a] + \gamma \sum_{s'} \mathbb{E}_\pi[R_{t+1} | s_{t+1} = s', s_t = s, a_t = a] P(s'|s,a)$$

$$\max_\pi = \mathbb{E}[r_{t+1} | s_t = s, a_t = a) + \gamma \sum_{s'} \mathbb{E}_\pi[R_{t+1} | s_{t+1} = s'] P(s'|s,a)$$

$$= \mathbb{E}[r_{t+1} | s_t = s, a_t = a] + \gamma \sum_{s',r} V^*(s') P(s',r|s,a)$$

Bellman Optimality Equation

$$V^*(s) = \max_a \sum_{r,s'} P(s',r|s,a) (r + \gamma V^*(s'))$$

$$Q^*(s,a) = \max_a \sum_{r,s'} P(s',r|s,a) (r + \gamma \max_{a'} Q^*(s',a'))$$

If we know $P(s',r|s,a)$

A system of $|S||A|$ equations with $|S||A|$ unknowns.

$$V^\pi(s') = \sum_a \pi(a|s) Q^\pi(s',a)$$

$$\underline{V^*(s') = \max_a} \; Q^*(s',a)$$

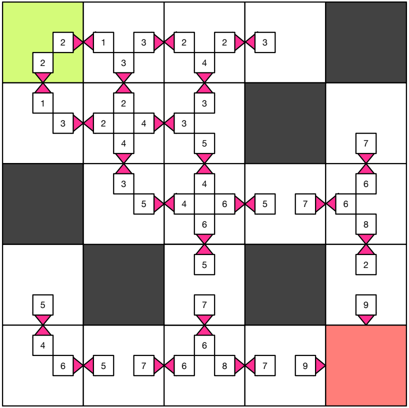
Optimal value function

The optimal value function for each state gives highest the expected return that can be obtained from that state.

2	3	4	3	
3	4	5		7
	5	6	7	8
5		7		9
6	7	8	9	10

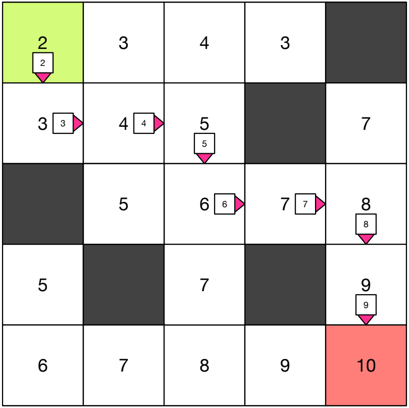
Optimal Q-function

The optimal Q-function for each state and action gives the highest expected return that can be obtained from that state when that action is taken.



Optimal policy

The optimal policy is the policy associated with the optimal value function or the optimal Q-function.



RL diagram:

① know $P(r_{t+1}=r', S_{t+1}=s' \mid S_t=s, a_t=a)$

Dynamic Programming:

i. policy iteration
- policy evaluation
  $V_{k+1}(s)=\sum_a \pi(a|s)\sum_{s',r} p(s',r|s,a)(r+\gamma V_k(s'))$
- policy improvement
  $\pi(a|s)=\arg\max_a Q_\pi(s,a)=\arg\max_a \sum_{s',r} p(s',r|s,a)(r+\gamma V_\pi(s'))$

ii. value iteration
- no policy involved
- optimal value fn. only
  $V_{k+1}(s)=\max_a \sum_{s',r} p(s',r|s,a)(r+\gamma V_k(s'))$

② unknown $P(r_{t+1}=r', S_{t+1}=s' \mid S_t=s, a_t=a)$

i. Monte Carlo prediction

simulate $V_\pi(s)$, $\forall s$

let's say n trajectories for each s.

now $\Pr\left[\frac{1}{n}\sum_{i=1}^{n}\hat{V}_\pi(s)-V_\pi(s)>\epsilon\right]\le e^{-\frac{2\epsilon^2}{nT^2}}$

if $V_\pi(s)\in[0,T]$ Hoeffding's Inequality

ii. now to do planning, we need to estimate $Q_\pi(s,a)$