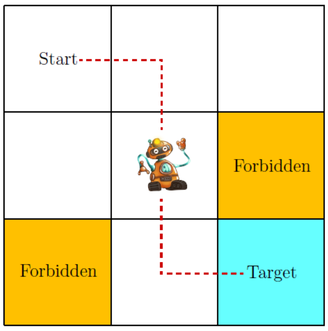
Lecture 1 Basic Concepts of Reinforcement Learning

—Rton

1. An Grid World Example

Firstly, let's consider the situation that there is a robot on start point and it's looking for the *target*, a special location in the map, there are somewhere are forbidden to reach and the robot will be punished if this happend. Showing as following figure.

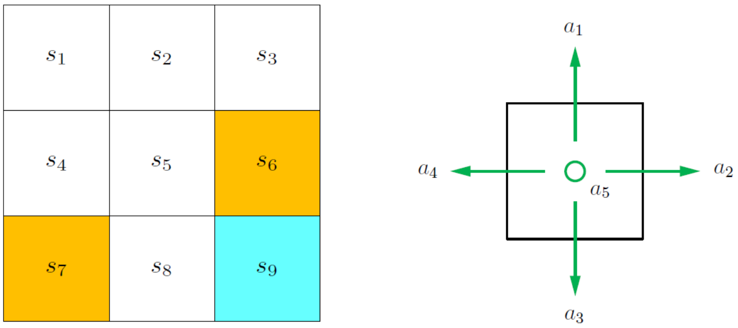


Therefore, there are some requerment or expectation about it, **we would like to decrease the all of distance from the start point to the target, and we want to avoid the robot don't have a collsion with the boundary or forbidden area.** Given this consideration, how to evalute the behavior of the robot with mathematic? what policy the robot take is the best? and then, how to calcute the best policy of the robot? As a matter of fact, the goal of reinforcement learning is to resolve how to get the best policy of the robot.

2. State and Action

From now on, we will start studying reinforcement learning and firstly we learn basic concepts. The first conception is *State*, which describes the agents(robots) status with respect to the environment. In the former grid example, what we consider is the location on the grid, so the state of robot is the robot location. Further more, we can know the robot state is belong to the set $S = \{s_1, s_2, \ldots, s_9\}$ since there are all nine cells. And the S is called the *State Space*.

The process of looking for the target need to change the state with continue. The another important thing is making the decision that how to move in current state. So we introduce the *Action*, for each state, the agent can take to do is called the *Action*. In the grid world example, the all possible action the agent can take including moving upward, moving rightward, moving downward, moving leftward and unchanged. These five action we can denote as a_1, a_2, a_3, a_4, a_5, the set of all action is called the *Action Space*, namely $A(S_i) = \{a_1, a_2, a_3, a_4, a_5\}$ (There is a explanation about why the action is represented $A(S_i)$, because the aciton depend on the current state, and the $A(S_i)$ means that the possible action in state S_i). Showing as following figure.



(a) States (b) Actions

3. State Transition

As above mentioned, the process of looking for the target is dynamic so the agent may move from one state to another state. The transition pocess is called *State Transition*, and we can express it as following.

$$s_1 \xrightarrow{a_2} s_2$$

There are two examples need to be considered, take the grid world example.

- If the agent take action a_1 at state s_1, what is the next state?
- If the agent take action a_2 at state s_5 and the forbidden area is unavailable, what is the next state?

So we shoud set some additional rules to ensure the process is logical, we define the following state transition.

$$s_1 \xrightarrow{a_1} s_1, s_5 \xrightarrow{a_2} s_5$$

Further more, we will introduce the *Conditional Probability* to describe the state transition. we discuss the deterministic transition firstly, which means the next state is known after a specific action at specific state. For instance, takin action a_2 at state s_1 we will certainly get state s_2, which can express as follows by probability.

$$p\left(s_2|s_1, a_2\right) = 1$$

which means the probability from state s_1 to s_2 by taking action a_2 equel to 1. Similarly, we can express the other situation.

$$\begin{aligned} p\left(s_2|s_2, a_5\right) &= 1 \\ p\left(s_8|s_2, a_2\right) &= 0 \\ p\left(s_9|s_8, a_2\right) &= 1 \end{aligned}$$

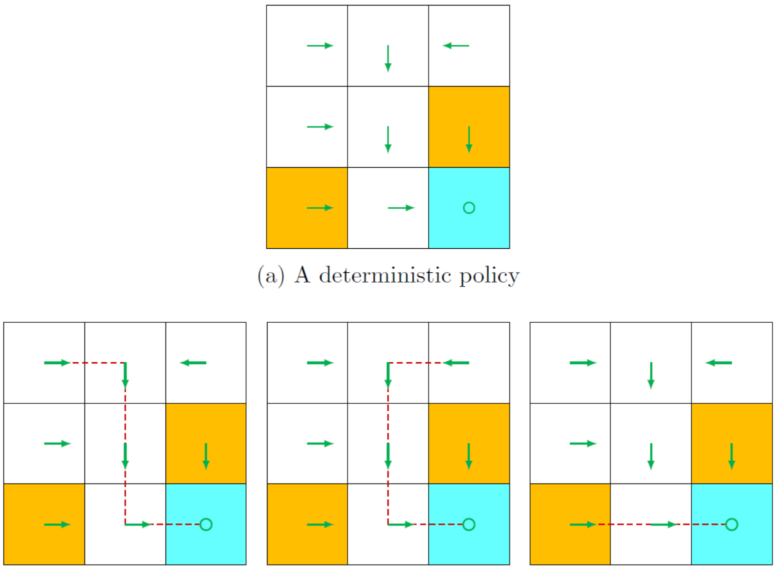
On the other hand, in many situation, the next state is unknown or uncertain after taking a specific ation at specific state, we call that is *stochastic*. For instance, the following equations represent the next state is uncertain

$$\begin{aligned} p\left(s_2|s_1, a_2\right) &= 0.5 \\ p\left(s_5|s_1, a_2\right) &= 0.5 \end{aligned}$$

So we not sure the agent location after it take action a_2 at state s_1.

4. Policy

The *Policy* is one of the best important conception in reinforcement learning, a *policy* tells the agent which method or action to take at every state. In other words, the *policy* represents the ability of judgement—should take which action at specific state. Take a grid world as exampel, the policy can be expressed as the following figure. Further more, we can deduce the trajectory that from any start point to the target point.



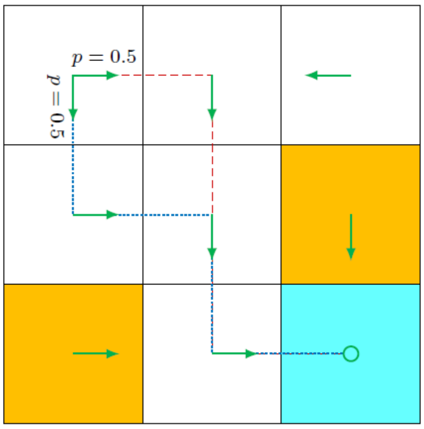
(a) A deterministic policy

(b) Trajectories obtained from the policy

As same as state transition, we can use the conditional probability to describe the policy mathematiclly. Denoting it as π and we can represent the policy with the following equation.

$$\begin{aligned}
\pi\left(a_1|s_1\right) &= 0\\
\pi\left(a_2|s_1\right) &= 1\\
\pi\left(a_3|s_1\right) &= 0\\
\pi\left(a_4|s_1\right) &= 0\\
\pi\left(a_5|s_1\right) &= 0
\end{aligned}$$

Mathematiclly, we can describe it with probability—the probability of the agent take action a_2 at state s_1 equal to 1. Obviously, this situation is a deterministic policy which means what the agent will take at every state is deterministic. Similarly, there are always some stochastic situation in general, action the agent will take is uncertain namely. There is a stochastic example it's policy is uncertain at some specific state.



Similarly, we can describe this situation with the following equation.

$$\begin{aligned}
\pi\left(a_1|s_1\right) &= 0\\
\pi\left(a_2|s_1\right) &= 0.5\\
\pi\left(a_3|s_1\right) &= 0.5\\
\pi\left(a_4|s_1\right) &= 0\\
\pi\left(a_5|s_1\right) &= 0
\end{aligned}$$

5. Reward

Reward is one of the most unique concepts in reinforcement learning. In above introduction, we can know what action should be take at every state based on the policy. The same important question is how to evaluate the each action is bad or good. Therefore, after excuting an action at a state, the agent obtains a reward, denoted as r, as the feedback from the environment. so the reward is a function of the state s and action a. Hence, it is also denoted as $r(s,a)$.

For instance, take the grid world example, the rewards are designed as follows: If the agent attempts to exit the boundary, let $r_{\text{boundary}} = -1$; If the agent attempts to enter a forbidden cell, let $r_{\text{forbidden}} = -1$; If the agent reaches the target state, let $r_{\text{target}} = +1$; Otherwise, $r_{\text{other}} = 0$.

Similarly, we can use conditional probability to describe it as follows.

$$\begin{aligned} p(r=-1|s_1,a_1) &= 1 \\ p(r\neq -1|s_1,a_1) &= 0 \end{aligned}$$

It should be pointed out that the reward is unrelated with the next state, A example is that if the agent obtains reward $r=1$ after taking action a_1 at state s_1 and then the agent enter next state s_2. so the reward r is related with the action a_1 and state s_1, unrelated with the state s_2.

6. Trajectories, Returns, and Episodes

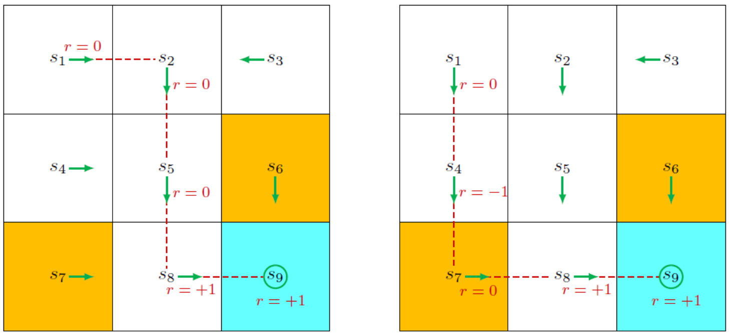
A *trajectory* is a state-action-reward chain. For example, this is a trajectory as follows.

$$s_1 \xrightarrow{a_1, r=0} s_2 \xrightarrow{a_3, r=0} s_5 \xrightarrow{a_3, r=0} s_8 \xrightarrow{a_2, r=1} s_9$$

A *return* of this trajectory is defined as the sum of all the rewards collected along the trajectory:

$$\text{return} = 0+0+0+1 = 1$$

The return can be used to evaluate the policy. For example, we can evaluate the two different policy as follows.



(a) Policy 1 and the trajectory

(b) Policy 2 and the trajectory

The left policy return can be calculated as 1 and the right policy return can be calculated as 0, so we can assert the left policy is better than the right policy.

Immediate and future rewards: A return consists of an immediate reward and future rewards. Here, the immediate reward is the reward obtained after taking an action at the initial state; the future rewards refer to the rewards obtained after leaving the initial state. It is possible that the immediate reward is negative while the future reward is positive. Thus, which actions to take should be determined by the return (i.e., the total reward) rather than the immediate reward to avoid short-sighted decisions.

In most of situation, the trajectory is defined for a infinite-length trajectory, return can also be infinite based on the above definition, like the follows.

$$s_1 \xrightarrow{a_1, r=0} s_2 \xrightarrow{a_3, r=0} s_5 \xrightarrow{a_3, r=0} s_8 \xrightarrow{a_2, r=1} s_9 \xrightarrow{a_5, r=1} s_9 \xrightarrow{a_5, r=1} s_9 \ldots$$

$$\text{return} = 0 + 0 + 0 + 1 + 1 + 1 + \cdots = \infty$$

The infinite value is very inconvenient in calculation process, in order to make the infinite tarjectory return can be calculated. we introduce the *discount return* concept for a infinite trajectory. In particular, the discounted return is the sum of the discounted rewards, it can be expressed as follows.

$$\text{return} = \gamma^0 0 + \gamma^1 0 + \gamma^2 0 + \gamma^3 1 + \gamma^4 1 + \gamma^5 1 + \cdots$$

where $\gamma \in (0, 1)$ is called the *discount rate*, if the γ is close to 0, then the agent places more emphasis on rewards obtained in the near time, the policy would be short-sighted; if the γ is close to 1, then the agent places more emphasis on the far futher rewards, the policy would is far-sighted. So the discount return can be calculated as:

$$\text{discounted return} = \gamma^3 (1 + \gamma + \gamma^2 + \cdots) = \gamma^3 \frac{1}{1 - \gamma}$$

One important notion that was not explicitly mentioned in the above discussion is theepisode. When interacting with the environment by following a policy, the agent may stopat some terminal states. The resulting trajectory is called an episode (or a trial). If the environment or policy is stochastic, we obtain episodes when starting from the same state.

7. Markov Decision Process(MDP)

An MDP is a general framework for describing stochastic dynamical systems. the mainly include the following key ingredients.

a) Sets

— State Space

— Action Space

— Reward Set

b) Model

— State transition probability

— Reward probability

c) Policy

d) Markov Property: The Markov property refers to the memoryless property of a stochastic process.

Finally, reinforcement learning can be described as an agent-environment interaction process. The agent is a decision-maker that can sense its state, maintain policies, and execute actions. Everything outside of the agent is regarded as the environment. In the grid world examples, the agent and environment correspond to the robot and grid world, respectively. After the agent decides to take an action, the actuator executes such a decision. Then, the state of the agent would be changed and a reward can be obtained. By using interpreters, the agent can interpret the new state and the reward. Thus, a closed loop can be formed.