Overview

1. Learning
2. Definition
3. Markov Decision Processes (MDP)
# Table of Contents

1. Learning
2. Definition
3. Markov Decision Processes (MDP)
Why do we need to learn?
Why do we need to learn?

There are (at least) two distinct reasons to learn:

1. Find previously unknown solutions. E.g., a program that can play Go better than any human, ever
Why do we need to learn?

There are (at least) two distinct reasons to learn:

1. Find previously unknown solutions. E.g., a program that can play Go better than any human, ever

2. Find solutions online, for unforeseen circumstances. E.g., a robot that can navigate terrains that differ greatly from any expected terrain
Why do we need to learn?

There are (at least) two distinct reasons to learn:

1. Find previously unknown solutions. E.g., a program that can play Go better than any human, ever

2. Find solutions online, for unforeseen circumstances. E.g., a robot that can navigate terrains that differ greatly from any expected terrain

Reinforcement learning seeks to provide algorithms for both cases

**Note** that the second point is not (just) about generalization — it is about learning efficiently online, during operation.
Why do we need to learn?

Science of learning to make decisions from interaction. This requires us to think about

- ...time
Why do we need to learn?

Science of learning to make decisions from interaction. This requires us to think about

- ...time
- ...(long-term) consequences of actions
Science of learning to make decisions from interaction. This requires us to think about

- ...time
- ...(long-term) consequences of actions
- ...actively gathering experience
Why do we need to learn?

Science of learning to make decisions from interaction. This requires us to think about

- ...time
- ...(long-term) consequences of actions
- ...actively gathering experience
- ...predicting the future
Why do we need to learn?

Science of learning to make decisions from interaction. This requires us to think about:

- ...time
- ...(long-term) consequences of actions
- ...actively gathering experience
- ...predicting the future
- ...dealing with uncertainty
Examples:  
- Fly a helicopter
Examples of decision problems

Examples:

- Fly a helicopter
- Manage an investment portfolio
Examples of decision problems

Examples:
- Fly a helicopter
- Manage an investment portfolio
- Control a power station
Examples of decision problems

Examples:

- Fly a helicopter
- Manage an investment portfolio
- Control a power station
- Make a robot walk
Examples of decision problems

Examples:

- Fly a helicopter
- Manage an investment portfolio
- Control a power station
- Make a robot walk
- Play video or board games

These are all reinforcement learning problems (no matter which solution method you use)
Table of Contents

1 Learning

2 Definition

3 Markov Decision Processes (MDP)
Core concepts of a reinforcement learning system are:

- Environment
Core concepts of a reinforcement learning system are:

- Environment
- Reward signal
Core concepts of a reinforcement learning system are:

- Environment
- Reward signal
- Agent, containing:
  - Agent state
  - Policy
  - Value function (probably)
  - Model (optionally)
The **agent** is acting in an **environment**. How the environment reacts to certain actions is defined by a **model** which we may or may not know. The agent can stay in one of many **states** \( s \in S \) of the environment, and choose to take one of many **actions** \( a \in A \) to switch from one state to another. Which state the agent will arrive in is decided by the **transition probabilities** between states \( P(s'|s,a) \). Once an action is taken, the environment delivers a **reward** \( r \in R \) as a feedback.
Table of Contents

1 Learning

2 Definition

3 Markov Decision Processes (MDP)
Markov Decision Processes (MDP)
Agent
Agents

Environment
Agent \rightarrow \mathbf{A}_t \rightarrow \text{Environment} \rightarrow \mathbf{S}_t \rightarrow \text{Agent}
Agent

Environment

$A_{t+1}$

$R_{t+1}$

$S_{t+1}$
Definition

Markov decision process (MDP) defined by the tuple $\langle S, A, P, R, \gamma \rangle$, where

- $S$ is the set of all states, which characterizes the configuration of the environment;
- $A$ denotes actions which the agent can take;
- $R$ is the reward function;
- $P$ is the states transition probability distribution;
- $\gamma \in (0, 1]$ is a discount factor.
Finite Markov Decision Processes (MDP)

Markov property:

\[ P[S_{t+1}|S_t] = P[S_{t+1}|S_1, S_2, \ldots, S_t] \]
Markov property:

\[ \mathbb{P}[S_{t+1} | S_t] = \mathbb{P}[S_{t+1} | S_1, S_2, \ldots, S_t] \]

“The future is independent of the past given the present”

Daily life trajectory:

\[ S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, \ldots, S_T \]
STATES (S)
Definition

Markov decision process (MDP) defined by the tuple \( \langle S, A, P, R, \gamma \rangle \), where
- \( S \) is the set of all states, which characterizes the configuration of the environment;
- \( A \) denotes actions which the agent can take;
- \( R \) is the reward function;
- \( P \) is the states transition probability distribution;
- \( \gamma \in (0, 1] \) is a discount factor.
Environment

Grid world 3x3

\[
\begin{array}{ccc}
S_1 & S_2 & S_3 \\
S_4 & S_5 & S_6 \\
S_7 & S_8 & S_9 \\
\end{array}
\]
Grid world 3x3

\[ S_t \in \{s_1, s_2, s_3, \ldots, s_9\} \]
ACTIONS (A)
Markov decision process (MDP) defined by the tuple $\langle S, A, P, R, \gamma \rangle$, where

- $S$ is the set of all states, which characterizes the configuration of the environment;
- $A$ denotes actions which the agent can take;
- $R$ is the reward function;
- $P$ is the states transition probability distribution;
- $\gamma \in (0, 1]$ is a discount factor.
Grid world 3x3

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
</tr>
</tbody>
</table>

Agent
Grid world 3x3

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1$</td>
<td>$s_2$</td>
<td>$s_3$</td>
</tr>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
</tr>
</tbody>
</table>

Agent

- UP
- LEFT
- RIGHT
- DOWN
### Grid World 3x3

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
</tr>
</tbody>
</table>

**Agent**

- UP
- DOWN
- LEFT
- RIGHT

$A_t \in \{\text{UP, DOWN, LEFT, RIGHT}\}$
Definition

Markov decision process (MDP) defined by the tuple \( \langle S, A, P, R, \gamma \rangle \), where

- **S** is the set of all states, which characterizes the configuration of the environment;
- **A** denotes actions which the agent can take;
- **\( R \) is the reward function**;
- **P** is the states transition probability distribution;
- \( \gamma \in (0, 1] \) is a discount factor.
## Rewards

<table>
<thead>
<tr>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$S_3$  +1</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Robot" /></td>
<td><img src="image2" alt="Heart" /></td>
<td><img src="image3" alt="Gold" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$S_4$</th>
<th>$S_5$ -5</th>
<th>$S_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Robot" /></td>
<td><img src="image4" alt="Monster" /></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$S_7$</th>
<th>$S_8$ +3</th>
<th>$S_9$ +10</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image5" alt="Diamond" /></td>
<td><img src="image2" alt="Heart" /></td>
<td><img src="image6" alt="Trophy" /></td>
</tr>
</tbody>
</table>
Rewards

\[ R_t \in \{-5, 0, +1, +3, +10\} \]
TRANSITION PROBABILITY (P)
Definition

Markov decision process (MDP) defined by the tuple $\langle S, A, P, R, \gamma \rangle$, where

- $S$ is the set of all states, which characterizes the configuration of the environment;
- $A$ denotes actions which the agent can take;
- $R$ is the reward function;
- $P$ is the states transition probability distribution;
- $\gamma \in (0, 1]$ is a discount factor.
Transition Probability

\[ p(s', r | s, a) \]
Transition Probability

\[ p(s', r | s, a) = P(S_{t+1} = s', R_{t+1} = r | S_t = s, A_t = a) \]
Transition Probability

$$p(s_2, 0|s_1, \text{RIGHT})$$
Transition Probability

\[
p(s_2, 0|s_1, \text{RIGHT}) = P(S_{t+1} = S_2, R_{t+1} = 0|S_t = s_1, A_t = \text{RIGHT})
\]
Transition Probability

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
</tr>
</tbody>
</table>

$p(s_2, 0|s_1, \text{RIGHT}) = P(S_{t+1} = s_2, R_{t+1} = 0|S_t = s_1, A_t = \text{RIGHT}) = 1$
Transition Probability

\[ p(s_2, 0 | s_1, \text{RIGHT}) = 1 \]
## Transition Probability

$$p(s_2, 0 | s_1, \text{RIGHT}) = 1$$

$$p(s_2, 0 | s_1, \text{LEFT}) =$$
### Transition Probability

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
<th>$s_4$</th>
<th>$s_5$</th>
<th>$s_6$</th>
<th>$s_7$</th>
<th>$s_8$</th>
<th>$s_9$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="#" alt="Robot" /></td>
<td><img src="#" alt="Heart" /></td>
<td>+1</td>
<td><img src="#" alt="Heart" /></td>
<td><img src="#" alt="Monster" /></td>
<td>-5</td>
<td><img src="#" alt="Monster" /></td>
<td><img src="#" alt="Diamond" /></td>
<td><img src="#" alt="Cup" /></td>
</tr>
</tbody>
</table>

\[
p(s_2, 0 \mid s_1, \text{RIGHT}) = 1
\]

\[
p(s_2, 0 \mid s_1, \text{LEFT}) = 0
\]

\[
p(s_8, 0 \mid s_1, \text{RIGHT}) = 
\]
Transition Probability

\[
p(s_2, 0|s_1, RIGHT) = 1
\]
\[
p(s_2, 0|s_1, LEFT) = 0
\]
\[
p(s_8, 0|s_1, RIGHT) = 0
\]
## Transition Probability

<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
<th>$+1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
<td>$-5$</td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
<td>$+10$</td>
</tr>
</tbody>
</table>

A) $p(s_4, +1|s_1, DOWN) = ?$

B) $p(s_6, 0|s_3, DOWN) = ?$

C) $p(s_9, +10|s_8, RIGHT) = ?$

D) $p(s_2, 0|s_3, LEFT) = ?$
DISCOUNT FACTOR ($\gamma$)
Markov decision process (MDP) defined by the tuple \( \langle S, A, P, R, \gamma \rangle \), where

- \( S \) is the set of all states, which characterizes the configuration of the environment;
- \( A \) denotes actions which the agent can take;
- \( R \) is the reward function;
- \( P \) is the states transition probability distribution;
- \( \gamma \in (0, 1] \) is a discount factor.
Discount Factor

The discounting factor $\gamma \in (0, 1]$ penalize the rewards in the future.

Reward at time $k$ worth only $\gamma^{k-1}$
Discount Factor: Motivation

- The future rewards may have higher uncertainty
Discount Factor: Motivation

- The future rewards may have higher uncertainty
- The future rewards do not provide immediate benefits (As human beings, we might prefer to have fun today rather than 5 years later ;)}
Discount Factor: Motivation

- The future rewards may have higher uncertainty
- The future rewards do not provide immediate benefits (As human beings, we might prefer to have fun today rather than 5 years later ;)
- Discounting provides mathematical convenience
Discount Factor: Motivation

- The future rewards may have higher uncertainty
- The future rewards do not provide immediate benefits (As human beings, we might prefer to have fun today rather than 5 years later ;)
- Discounting provides mathematical convenience
- It is sometimes possible to use undiscounted Markov reward processes (i.e. $\gamma = 1$) e.g. if all sequences terminate
RETURN (G or R)
RL agents learn to maximize discounted cumulative future reward ($R$).
Cumulative reward:

\[ R_t = r_{t+1} + r_{t+2} + r_{t+3} + r_{t+4} + \cdots = \sum_{k=0}^{\infty} r_{t+k+1} \]
Cumulative reward:

\[ R_t = r_{t+1} + r_{t+2} + r_{t+3} + r_{t+4} + \cdots = \sum_{k=0}^{\infty} r_{t+k+1} \]

**Discounted cumulative reward:**

\[ R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 r_{t+4} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \]

where \( 0 \leq \gamma \leq 1 \)
The \textit{return} $G_t$ or $R_t$ is the total discounted reward from time-step $t$.

$$G_t = R_t = r_{t+1} + \gamma r_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

- $\gamma$ is a discount factor ($\gamma \in [0, 1)$)
- $r$ is the immediate reward, $R$ is the cumulative reward
- The value of receiving reward $R$ after $k + 1$ time-steps is $\gamma^k R$
<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
<th>$s_4$</th>
<th>$s_5$</th>
<th>$s_6$</th>
<th>$s_7$</th>
<th>$s_8$</th>
<th>$s_9$</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Robot]</td>
<td>![Monster]</td>
<td>![Gold Coin]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>![Diamond]</td>
<td>![Trophy]</td>
</tr>
</tbody>
</table>

Episode 1: $(s_1, \text{RIGHT}, 0, s_2, \text{RIGHT}, +1, s_3, \text{DOWN}, 0, s_6, \text{DOWN}, +10)$
Episode 1: $(s_1, \text{RIGHT}, 0, s_2, \text{RIGHT}, +1, s_3, \text{DOWN}, 0, s_6, \text{DOWN}, +10)$

$\gamma = 0.5$

$G_{\text{Episode1}}^1 = 0 + 0.5 \times 1 + 0.5^2 \times 0 + 0.5^3 \times 10 = 1.75$
<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
<th>+1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
<td>-5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
<td>+3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+10</td>
</tr>
</tbody>
</table>

Episode 2: (\(s_1, \text{DOWN}, 0, s_4, \text{RIGHT}, -5, s_5, \text{RIGHT}, 0, s_6, \text{DOWN}, +10\))
Episode 2: \((s_1, \text{DOWN}, 0, s_4, \text{RIGHT}, -5, s_5, \text{RIGHT}, 0, s_6, \text{DOWN}, +10)\)

\[ \gamma = 0.5 \]

\[ G_1^{\text{Episode2}} = \]
Episode 2: \((s_1, \text{DOWN}, 0, s_4, \text{RIGHT}, -5, s_5, \text{RIGHT}, 0, s_6, \text{DOWN}, +10)\)

\[\gamma = 0.5\]

\[G_1^{\text{Episode2}} = 0 + 0.5 \times (-5) + 0.5^2 \times 0 + 0.5^3 \times 10 = -1.25\]
<table>
<thead>
<tr>
<th>$s_1$</th>
<th>$s_2$</th>
<th>$s_3$</th>
<th>+1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_4$</td>
<td>$s_5$</td>
<td>$s_6$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_7$</td>
<td>$s_8$</td>
<td>$s_9$</td>
<td>+10</td>
</tr>
<tr>
<td></td>
<td>+3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Episode 3: $(s_1, \text{DOWN}, 0, s_4, \text{DOWN}, 0, s_7, \text{RIGHT}, +3, s_8, \text{RIGHT}, +10)$
Episode 3: \((s_1, DOWN, 0, s_4, DOWN, 0, s_7, RIGHT, +3, s_8, RIGHT, +10)\)

\[
\gamma = 0.5
\]

\[
G^\text{Episode3}_1 = ?
\]
POLICIES (π)
**Definition**

A *policy* $\pi$ is a distribution over actions given states. It defines the agent’s behaviour. It can be either deterministic or stochastic:

- **Deterministic:** $\pi(s) = a$
- **Stochastic:** $\pi(a|s) = \mathbb{P}_\pi[A = a|S = s]$

- A policy fully defines the behaviour of an agent
- MDP policies depend on the current state (not the history)
What is the optimal* policy for our agent?
Summary So Far

- Mains reasons to learn (not just for agents) are to find previously unknown solutions and to find solutions in unforeseen circumstances
- Core parts of a reinforcement learning are: Environment, Reward, Agent
- Markov property: The future is independent of the past given the present
- The discounting factor $\gamma \in (0,1)$ penalize the rewards in the future
- Markov Decision Process (MDP) defined as a tuple $\langle S, A, P, R, \gamma \rangle$