



# [S&B Book] Chapter 5: Monte Carlo Methods

Tags

- In this chapter, we do not assume complete knowledge of the environment;
- **Monte Carlo methods** are ways of solving the reinforcement learning problem based on averaging sample returns. To ensure well-defined returns are available, we define Monte Carlo methods only for episodic tasks. [Only on the completion of an episode are value estimates and policies changes.](#)

## ▼ 5.1 Monte Carlo Prediction

- Expected return - expected cumulative future discounted reward starting from that state.
- As more returns are observed, the average should converge to the expected value. This idea underlies all Monte Carlo methods.
- A [visit](#) to  $s$  - each occurrence of state  $s$  in an episode is called a *visit* to  $s$ .
- **First-visit and every-visit MC method:**

In an episode,  $s$  may be visited multiple times:

1. First-visit

The first time  $s$  is visited in an episode. The first-visit MC method estimates  $v_\pi(s)$  as the average of the returns following first visits to  $s$ .

2. Every-visit:

The every-visit MC method averages the returns following all visits to  $s$ .

- First-visit MC prediction algorithm

### First-visit MC prediction, for estimating $V \approx v_\pi$

Input: a policy  $\pi$  to be evaluated

Initialize:

$V(s) \in \mathbb{R}$ , arbitrarily, for all  $s \in \mathcal{S}$

$Returns(s) \leftarrow$  an empty list, for all  $s \in \mathcal{S}$

Loop forever (for each episode):

Generate an episode following  $\pi$ :  $S_0, A_0, R_1, S_1, A_1, R_2, \dots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode,  $t = T-1, T-2, \dots, 0$ :

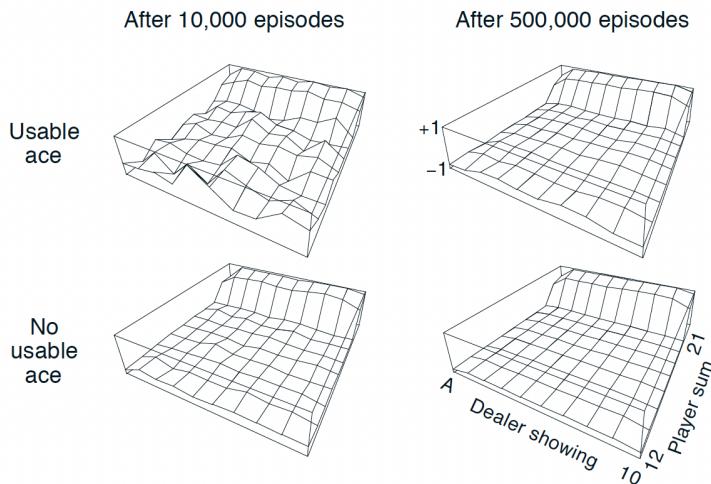
$G \leftarrow \gamma G + R_{t+1}$

Unless  $S_t$  appears in  $S_0, S_1, \dots, S_{t-1}$ :

Append  $G$  to  $Returns(S_t)$

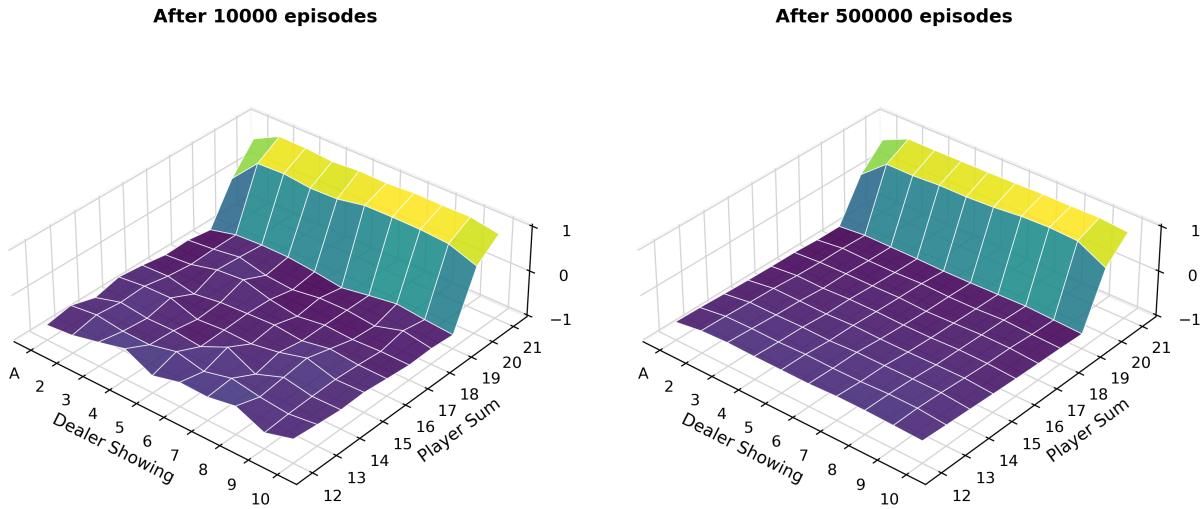
$V(S_t) \leftarrow \text{average}(Returns(S_t))$

#### Example 5.1: Blackjack



#### Implementation:

Due to the limitation of the Gymnasium Blackjack environment, the "no usable case" is temporarily not implemented, but will be updated in the future.



[Reinforcement-Learning-2nd-Edition-Notes-Codes/chapter\\_05\\_monte\\_carlo\\_methods/example\\_5\\_1\\_blackjack.py](https://github.com/terrence-ou/Reinforcement-Learning-2nd-Edition-Notes-Codes/blob/main/chapter_05_monte_carlo_methods/example_5_1_blackjack.py) at main · terrence-ou/Reinforcement-Learning-2nd-Edition-Notes-Codes · GitHub

Notes and code implementations of examples and algorithms of the book Reinforcement Learning, 2nd Edition - Reinforcement-Learning-2nd-Edition-Notes-Codes/chapter\_05\_monte\_carlo\_methods/example\_5\_1...

[https://github.com/terrence-ou/Reinforcement-Learning-2nd-Edition-Notes-Codes/blob/main/chapter\\_05\\_monte\\_carlo\\_methods/example\\_5\\_1\\_blackjack.py](https://github.com/terrence-ou/Reinforcement-Learning-2nd-Edition-Notes-Codes/blob/main/chapter_05_monte_carlo_methods/example_5_1_blackjack.py)