Implementation: MC Prediction (State Values)

The pseudocode for (first-visit) MC prediction (for the state values) can be found below. (*Feel free to implement either the first-visit or every-visit MC method. In the game of Blackjack, both the first-visit and every-visit methods return identical results.*)

<div style="border:2px solid #29b6d8; padding:1em;">

<div style="background:#29b6d8; color:white; text-align:center; font-weight:bold;">First-Visit MC Prediction (for State Values)</div>

Input: policy π, positive integer $num\_episodes$
Output: value function V ($\approx v_\pi$ if $num\_episodes$ is large enough)
Initialize $N(s) = 0$ for all $s \in \mathcal{S}$
Initialize $returns\_sum(s) = 0$ for all $s \in \mathcal{S}$
for $i \leftarrow 1$ **to** $num\_episodes$ **do**
$\quad$ Generate an episode $S_0, A_0, R_1, \ldots, S_T$ using π
$\quad$ **for** $t \leftarrow 0$ **to** $T - 1$ **do**
$\qquad$ **if** S_t *is a first visit (with return G_t)* **then**
$\qquad\quad$ $N(S_t) \leftarrow N(S_t) + 1$
$\qquad\quad$ $returns\_sum(S_t) \leftarrow returns\_sum(S_t) + G_t$
$\quad$ **end**
end
$V(s) \leftarrow returns\_sum(s)/N(s)$ for all $s \in \mathcal{S}$
return V

</div>

If you are interested in learning more about the difference between first-visit and every-visit MC methods, you are encouraged to read Section 3 of this paper. Their results are summarized in Section 3.6. The authors show:

- Every-visit MC is biased, whereas first-visit MC is unbiased (see Theorems 6 and 7).
- Initially, every-visit MC has lower mean squared error (MSE), but as more episodes are collected, first-visit MC attains better MSE (see Corollary 9a and 10a, and Figure 4).

Both the first-visit and every-visit method are **guaranteed to converge** to the true value function, as the number of visits to each state approaches infinity. (*So, in other words, as long as the agent gets enough experience with each state, the value function*

≡   Implementation

the **textbook**.

Please use the next concept to complete **Part 0: Explore BlackjackEnv** and **Part 1: MC Prediction: State Values** of `Monte_Carlo.ipynb`. Remember to save your work!

If you'd like to reference the pseudocode while working on the notebook, you are encouraged to open **this sheet** in a new window.

Feel free to check your solution by looking at the corresponding sections in `Monte_Carlo_Solution.ipynb`.

NEXT