3.4 MONTE-CARLO LEARNING

Il metodo *Monte-Carlo* (MC) ha lo stesso obiettivo della DP, ovvero stimare la *Value Function* e individuare la *policy ottimale*. Poiché non utilizza un modello dell'ambiente, questo metodo viene chiamato *model-free*.

Nello specifico il MC si occupa di stimare la *Value Function* calcolando la media dei *return* G_t ottenuti attraverso l'esperienza acquisita dall'agente in ogni episodio.

Dato che anche in questo caso è possibile utilizzare il concetto di *Policy Iteration*, si ritiene opportuno adottare la medesima suddivisione proposta in precedenza (Sezione 3.3).

3.4.1 Policy Evaluation (MC Prediction)

Nella prima fase l'obiettivo è quello di valutare $v_\pi(s)$. Pertanto, per ogni stato s che appare all'interno dell'episodio:

1. viene incrementato un contatore $N(s)$

$$N(s) \longleftarrow N(s) + 1 \tag{3.24}$$

2. viene memorizzato il *return* totale dello stato s

$$S(s) \longleftarrow S(s) + G_t \tag{3.25}$$

3. viene effettuata la media ottenendo così $V(S)$

$$V(s) = S(s)/N(s) \tag{3.26}$$

   E sulla base della legge dei grandi numeri:
   $V(s) \rightarrow v_\pi(s)$ quando $N(s) \rightarrow \infty$

Alla fine di ogni episodio viene aggiornata la funzione $V(s)$:

$$V(S_t) \leftarrow V(S_t) + \frac{1}{N(S_t)}(G_t - V(S_t)) \tag{3.27}$$

Nel caso di non-stazionarietà, ovvero processi in cui la distribuzione di probabilità cambia nel tempo, l'aggiornamento da effettuare è il seguente:

$$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t)) \tag{3.28}$$

Tuttavia, siccome il modello non risulta disponibile, si ritiene più utile stimare la *Action-Value Function* rispetto alla *State-Value Function* come segue:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha(G_t - Q(S_t, A_t)) \tag{3.29}$$

3.4.2 Policy Improvement

La fase di *policy improvement* consiste nell'individuare una *policy* migliore. In precedenza veniva utilizzato un'approccio di tipo *greedy*, ma poiché questo genera il cosiddetto *exploration and exploitation dilemma*, si è preferita la tecnica ϵ-*greedy* per risolvere questa criticità.

In questo metodo tutte le m azioni vengono provate con probabilità diversa da zero. Nello specifico; con probabilità $1 - \epsilon$ si sceglierà un azione *greedy*, mentre con probabilità ϵ si sceglierà un azione causale.

$$\pi(a|s) = \begin{cases} \epsilon/m + 1 - \epsilon & \text{if } a^* = \underset{a \in \mathcal{A}}{\text{argmax}} Q(s, a) \\ \epsilon/m & \text{altrimenti} \end{cases} \tag{3.30}$$

L'utilizzo della tecnica ϵ-*greedy* permette di definire un teorema il cui enunciato è:

Per ogni ϵ-*greedy policy* π, la ϵ-*greedy policy* π' si rivela migliore in riferimento al parametro q, ossia si ottiene che $v_{\pi'}(s) \geqslant v_{\pi}(s)$.

3.4.3 Monte-Carlo Policy Iteration (Control)

Come nella DP, anche nella MC la *Policy Iteration* consiste nell'alternarsi di *Policy Evaluation* e *Policy Improvement*. In Figura 3.3 viene descritto il funzionamento.

$$\pi_0 \xrightarrow{E} q_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} q_{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} \cdots \xrightarrow{I} \pi_* \xrightarrow{E} q_*$$

Figura 3.3: Funzionamento *Policy Iteration Monte-Carlo*

3.4.4 Limiti Monte-Carlo

Nonostante questa tecnica si presenti notevolmente più efficace rispetto alla DP, possiede dei limiti: apprende solamente da episodi che terminano e che vengono completati, e deve attendere la fine di un episodio per elaborare il valore della *Value Function*.