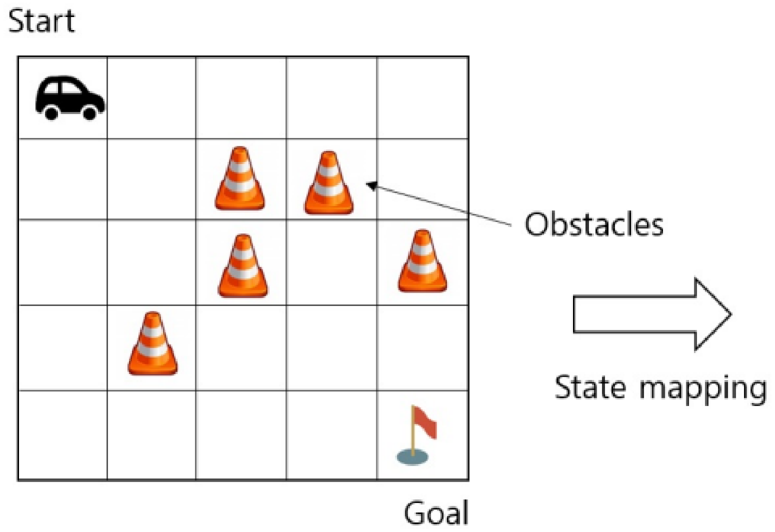
Monte Carlo Method

Presenter : Jongsoo Lee

Index

1. Introduction



1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Q-Bellman equation for π

$$
\begin{aligned}
Q^{\pi}(s,a) =& R(s,a) + \gamma E\left[Q^{\pi}(s_{k+1},a_{k+1}) | s_k = s, a_k = a, \pi\right] \\
=& R(s,a) + \gamma \sum_{a' \in A} \sum_{s' \in S} \pi(a'|s')P(s'|s,a)Q^{\pi}(s',a') \\
& s \in S, a \in A
\end{aligned}
$$

2. Background

1) Law of Large Number

2) Incremental Mean

3) Importance Sampling

2. Background – Law of Large Number

Theorem 2.7 (Weak law of large numbers) *Suppose that $X_1, X_2, \ldots$ is an infinite sequence of i.i.d. (Lebesgue integrable) random variables with expected value*

$$E[X_1] = E[X_2] = \cdots = \mu$$

Then, the sample average

$$\bar{X}_n := \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$$

converges in probability to the expected value

$$\bar{X}_n \xrightarrow{\mathrm{P}} \mu$$

as $n \to \infty$, that is, for any positive number ε

$$\boxed{\lim_{n\to\infty} P\left[|\bar{X}_n - \mu| > \varepsilon\right] = 0}$$

2. Background – Law of Large Number

Theorem 2.8 (Strong law of large numbers) *Suppose that* $X_1, X_2, \ldots$ *is an infinite sequence of i.i.d. (Lebesgue integrable) random variables with expected value*

$$E[X_1] = E[X_2] = \cdots = \mu$$

Then, the sample average

$$\bar{X}_n := \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$$

converges almost surely to the expected value

$$\bar{X}_n \overset{\text{a.s.}}{\to} \mu$$

as $n \to \infty$*, that is,*

$$\boxed{P\left[\lim_{n\to\infty} \bar{X}_n = \mu\right] = 1}$$

2. Background – Empirical Mean

Example 2.1 *Suppose that $X_1, X_2, \ldots$ is an infinite sequence of i.i.d. random variables with expected value*

$$E[X_1] = E[X_2] = \cdots = \mu$$

Then, the sample average

$$\bar{X}_n := \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$$

approximates μ, i.e., $\bar{X}_n \cong \mu$. Moreover, as $n \to \infty$, the sample average becomes closer to μ.

2. Background – Empirical Mean

The idea is to take an empirical mean with N number of episodes.

$$\begin{aligned}
\text{episod1} &= (s_0^{(1)}, a_0^{(1)}, r_0^{(1)}, s_1^{(1)}, a_1^{(1)}, r_1^{(1)}, \ldots, s_{\tau^{(1)}-1}^{(1)}, a_{\tau^{(1)}-1}^{(1)}, r_{\tau^{(1)}-1}^{(1)}, s_{\tau^{(1)}}^{(1)}) \\
\text{episod2} &= (s_0^{(2)}, a_0^{(2)}, r_0^{(2)}, s_1^{(2)}, a_1^{(2)}, r_1^{(2)}, \ldots, s_{\tau^{(2)}-1}^{(2)}, a_{\tau^{(2)}-1}^{(2)}, r_{\tau^{(2)}-1}^{(2)}, s_{\tau^{(2)}}^{(2)}) \\
&\ldots \\
\text{episodN} &= (s_0^{(N)}, a_0^{(N)}, r_0^{(N)}, s_1^{(N)}, a_1^{(N)}, r_1^{(N)}, \ldots, s_{\tau^{(N)}-1}^{(N)}, a_{\tau^{(N)}-1}^{(N)}, r_{\tau^{(N)}-1}^{(N)}, s_{\tau^{(N)}}^{(N)})
\end{aligned}$$

where we assume that the episodes are independent of each other and $s_0^{(1)} = s_0^{(2)} = \cdots = s_0^{(N)} = s$. We can take an average as follows:

$$\boxed{V^\pi(s) \cong \frac{1}{N} \sum_{i=1}^{N} G_0^{(i)}}$$

where

$$G_k^{(i)} = \sum_{t=k}^{\tau^{(i)}-1} \gamma^{t-k} r_t^{(i)}$$

2. Background – Incremental Mean

$$
\begin{aligned}
\mu_k &= \frac{1}{k}\sum_{j=1}^{k} x_j \\
&= \frac{1}{k}\left(x_k + \sum_{j=1}^{k-1} x_j\right) \\
&= \frac{1}{k}(x_k + (k-1)\mu_{k-1}) \\
&= \mu_{k-1} + \frac{1}{k}(x_k - \mu_{k-1})
\end{aligned}
$$

2. Background – Unbiased Estimator

$$E(\hat{\theta}) = \theta + bias(\theta)$$

2. Background – Importance Sampling

Estimate one distribution by sampling from another distribution

$$E_{x\sim p}[f(x)] \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i) \qquad \text{where } x_i \sim p.$$

$$\begin{aligned}
E_{x\sim p}[f(x)] &= \sum p(x)f(x) \\
&= \sum \frac{p(x)}{q(x)} q(x)f(x) \\
&= E_{x\sim q}\left[\frac{p(x)}{q(x)} f(x)\right] \\
&\approx \frac{1}{N}\sum_{i=1}^{N} \frac{p(x_i)}{q(x_i)} f(x_i) \qquad \text{where } x_i \sim q.
\end{aligned}$$

2. Background – Importance Sampling

$$E_{x \sim p}[f(x)] = E_{x \sim q}\left[\frac{p(x)}{q(x)} f(x)\right]$$

Variance of the original expectation:

$$Var_{x \sim p}[f(x)] = E_{x \sim p}[f(x)^2] - (E_{x \sim p}[f(x)])^2$$

Variance of the modified expectation:

$$\begin{aligned} Var_{x \sim q}\left[\frac{p(x)}{q(x)} f(x)\right] &= E_{x \sim q}\left[\left(\frac{p(x)}{q(x)} f(x)\right)^2\right] - \left(E_{x \sim q}\left[\frac{p(x)}{q(x)} f(x)\right]\right)^2 \\ &= E_{x \sim p}\left[\frac{p(x)}{q(x)} f(x)^2\right] - (E_{x \sim q}[f(x)])^2 \end{aligned}$$

3. Monte Carlo Learning

1. Monte Carlo Prediction

2. Monte Carlo Control

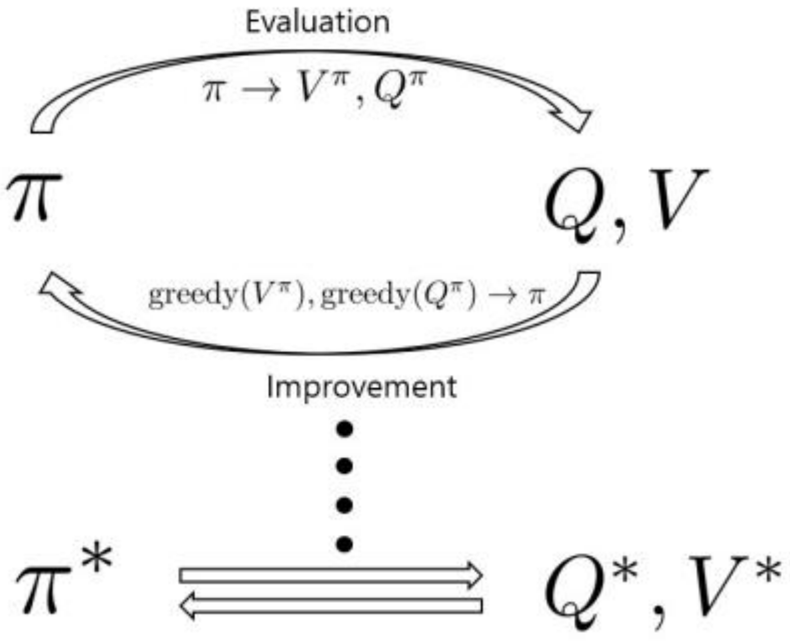


Figure 1.15: Policy iteration

3. Monte Carlo Prediction

From a set of episodes (large number of episodes), the value function $v_\pi(s)$ for each state s is estimated as follows.

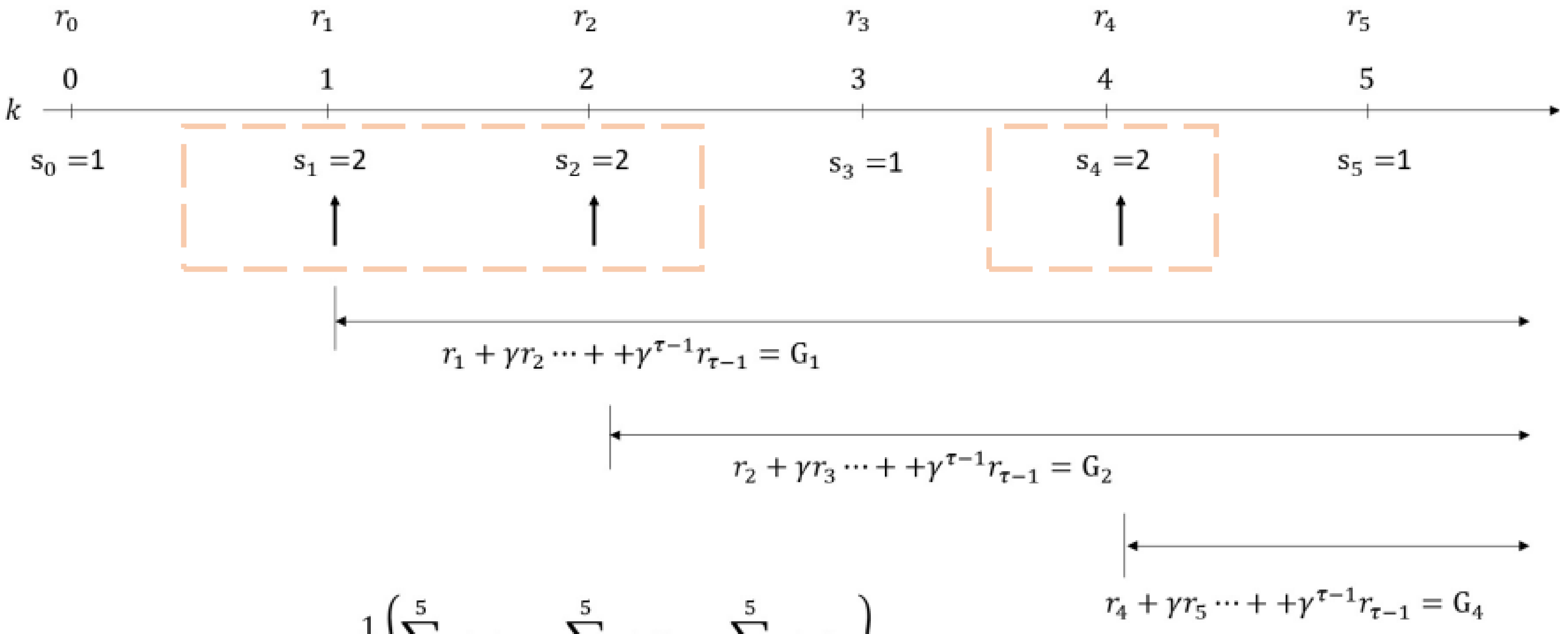
- Consider all episode one by one
- For each episode, find every time-step t that state s is visited.
- Increase the counter $N(s) \leftarrow N(s) + 1$
- Add new return $S(s) \leftarrow S(s) + G_t$
- After all episode are considered, we compute the estimated value function

$$V(s) = S(s)/N(s)$$

- By law of large numbers $V(s) \rightarrow v_\pi(s)$ as $N(s) \rightarrow \infty$.

time:	1	2	3	4
ep 1:	s_1, r_1,	s_2, r_2,	s_3, r_3,	s_4, r_4
ep 2:	s_2, r_2,	s_1, r_1,	s_4, r_4	
ep 3:	s_3, r_3,	s_1, r_1,	s_3, r_3,	s_4, r_4
ep 4:	s_2, r_4,	s_4, r_4		

3. Monte Carlo Prediction



$$V^{\pi}(2)\approx\frac{1}{3}\left\{\sum_{t=1}^{5}\gamma^{t-1}r_t+\sum_{t=2}^{5}\gamma^{t-2}r_t+\sum_{t=4}^{5}\gamma^{t-4}r_t\right\}$$

Figure 2.11: Every-visit Monte Calro

$$\boxed{V^{\pi}(s)\cong\frac{1}{K}(G_{k_1}+G_{k_2}+\cdots+G_{k_K})}$$

3. Monte Carlo Prediction

Algorithm 2 Every-visit Monte Calro prediction(recursive version)

```
Input: a policy π to be evaluated
Initialize
V(s) = 0 for all s ∈ S
m(s) = 0 for all s ∈ S
for i ∈ {0, 1, ...} do
    Generate an episode following π: (s_0, a_0, r_0, s_1, a_1, r_1, ..., s_{τ−1}, a_{τ−1}, r_{τ−1}, s_τ)
    G ← 0
    for k = τ − 1, τ − 2, ..., 0 do
        G ← γG + r_k
        m(s_k) ← m(s_k) + 1
        V(s_k) ← V(s_k) + 1/m(s_k) (G − V(s_k))
    end for
end for
```

Remark 2.2 *Let* $\tau = 5$.

1. $k = \tau - 1 = 4$: $G \leftarrow \gamma G + r_4 = r_4$
2. $k = \tau - 2 = 3$: $G \leftarrow \gamma G + r_3 = r_3 + \gamma r_4$
3. $k = \tau - 3 = 2$: $G \leftarrow \gamma G + r_2 = r_2 + \gamma r_3 + \gamma^2 r_4$
4. $k = \tau - 4 = 1$: $G \leftarrow \gamma G + r_1 = r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 r_4$

3. Monte Carlo Prediction

From a set of episodes (large number of episodes), the value function $v_\pi(s)$ for each state s is estimated as follows.

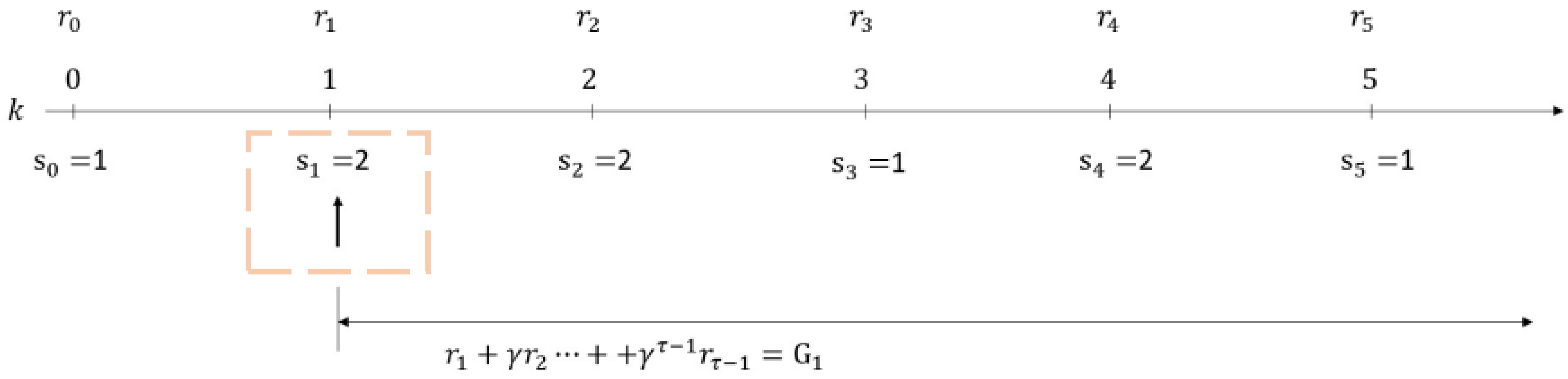
- Consider all episode one by one
- For each episode, find the first time-step t that state s is visited.
- Increase the counter $N(s) \leftarrow N(s) + 1$
- Add new return $S(s) \leftarrow S(s) + G_t$
- After all episode are considered, we compute the estimated value function

$$V(s) = S(s)/N(s)$$

- By law of large numbers $V(s) \to v_\pi(s)$ as $N(s) \to \infty$.

time:	1		2		3		4	
ep 1:	s_1,	r_1,	s_2,	r_2,	s_3,	r_3,	s_4,	r_4
ep 2:	s_2,	r_2,	s_1,	r_1,	s_4,	r_4		
ep 3:	s_3,	r_3,	s_1,	r_1,	s_3,	r_3,	s_4,	r_4
ep 4:	s_2,	r_4,	s_4,	r_4				

3. Monte Carlo Prediction



$$V^{\pi}(2) \approx \sum_{t=1}^{5} \gamma^{t-1} r_t$$

Figure 2.12: First-visit Monte Calro

3. Monte Carlo Prediction

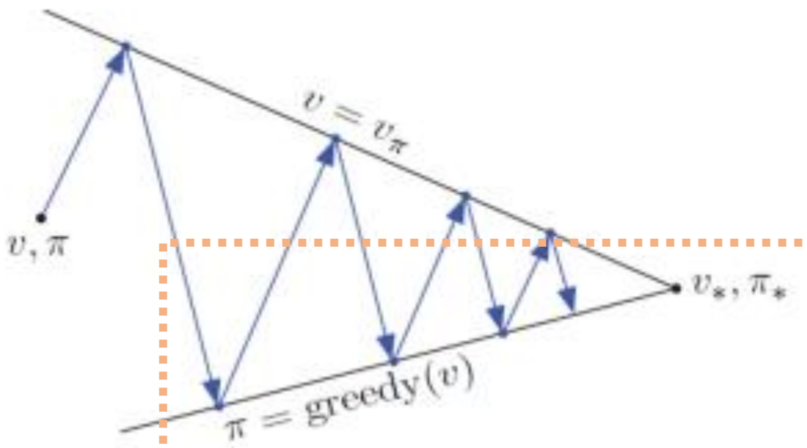
Algorithm 4 First-visit Monte Calro prediction (recursive version)

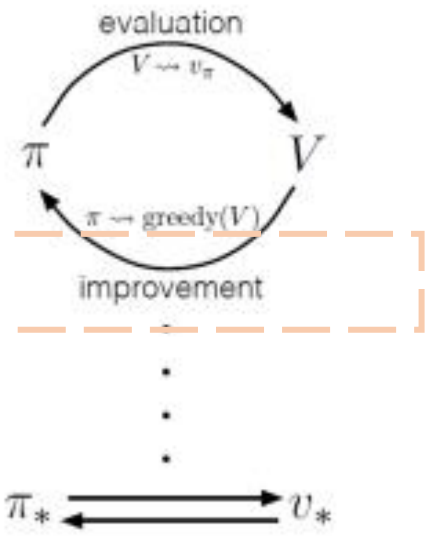
Input: a policy π to be evaluated
Initialize
$V(s) = 0$ for all $s \in S$
$Returns(s) \leftarrow$ an empty list for all $s \in S$
$m(s) = 0$ for all $s \in S$
for $i \in \{0, 1, \dots\}$ **do**
    Generate an episode following π: $(s_0, a_0, r_0, s_1, a_1, r_1, \dots, s_{\tau-1}, a_{\tau-1}, r_{\tau-1}, s_\tau)$
    $G \leftarrow 0$
    for $k = \tau - 1, \tau - 2, \dots, 0$ **do**
        $G \leftarrow \gamma G + r_k$
        if s_k does not appear in $s_0, s_1, \dots, s_{k-1}$ **then**
            $m(s_k) \leftarrow m(s_k) + 1$
            $V(s_k) \leftarrow V(s_k) + \frac{1}{m(s_k)}(G - V(s_k))$
        end if
    end for
end for

3. Monte Carlo Prediction

	Every Visit Monte Carlo	First Visit Monte Carlo
Advantage	Sample Efficiency	Higher Quality Estimation (Less reuse of data)
Disadvantage	Low Quality Estimation (Too many reuse of the same data)	Sample Inefficiency

3. Monte Carlo Control





$$\pi'(s) = \arg\max_a \left(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a V(s') \right)$$

$$\begin{aligned}
\pi'(s) &= \arg\max_a q_\pi(s, a) \\
&= \arg\max_a E_\pi[R_{t+1} + \gamma v_\pi(S_{t+1}) | S_t = s, A_t = a] \\
&= \arg\max_a \left(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a v_\pi(s') \right)
\end{aligned}$$

3. Monte Carlo Control – Action Selection Method

The exploration and exploitation dilemma

- Exploitation: to learn V^π or Q^π for given π, we need to 'exploit' π to generate episodes.
- Exploration: to learn Q^π for given π, we need to 'explore' every actions $a \in A$ at every $s \in S$.

Action Selection Method

1) Random Policy
2) Greedy Policy
3) Soft Greedy Policy
4) Boltzmann Approach
5) Bayesian Approach

3. Monte Carlo Control – Soft greedy Policy

ε-soft policy ($\varepsilon \in (0,1)$): it converts a deterministic policy into an approximate stochastic policy as follows:

ε-soft policy associated with a given π

$$\begin{cases} \text{Choose } a \in A \text{ randomly from } A \text{ with probability } \varepsilon \\ \text{Choose } a = \pi(s) \text{ with probability } 1 - \varepsilon \end{cases}$$

which leads to the equivalent stochastic policy

ε-soft policy associated with a given π

$$\pi_\varepsilon(a|s) = \begin{cases} 1 - \varepsilon + \frac{\varepsilon}{|A|} & \text{if} \quad a = \pi(s) \\ \frac{\varepsilon}{|A|} & \text{if} \quad a \neq \pi(s) \end{cases}$$

3. Monte Carlo Control

Algorithm 6 First-visit Monte Calro method (batch version) for Q-function with ε-soft policy

Input: a policy π to be evaluated
Initialize
$Q(s, a) = 0$ for all $s \in S$ and $a \in A$
$Returns(s, a) \leftarrow$ an empty list for all $s \in S$ and $a \in A$.
for $i \in \{0, 1, \ldots\}$ **do**
    Generate an episode following π_ε: $(s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_{\tau-1}, a_{\tau-1}, r_{\tau-1}, s_\tau)$
    $G \leftarrow 0$
    for $k = \tau - 1, \tau - 2, \ldots, 0$ **do**
        $G \leftarrow \gamma G + r_k$
        if the pair (s_k, a_k) does not appear in $(s_0, a_0), (s_1, a_1), \ldots, (s_{k-1}, a_{k-1})$ **then**
            Append G to the list $Rerturns(s_k, a_k)$
            $Q(s_k, a_k) \leftarrow$ average$(Returns(s_k, a_k))$
        end if
    end for
end for

4. Off-Policy Learning

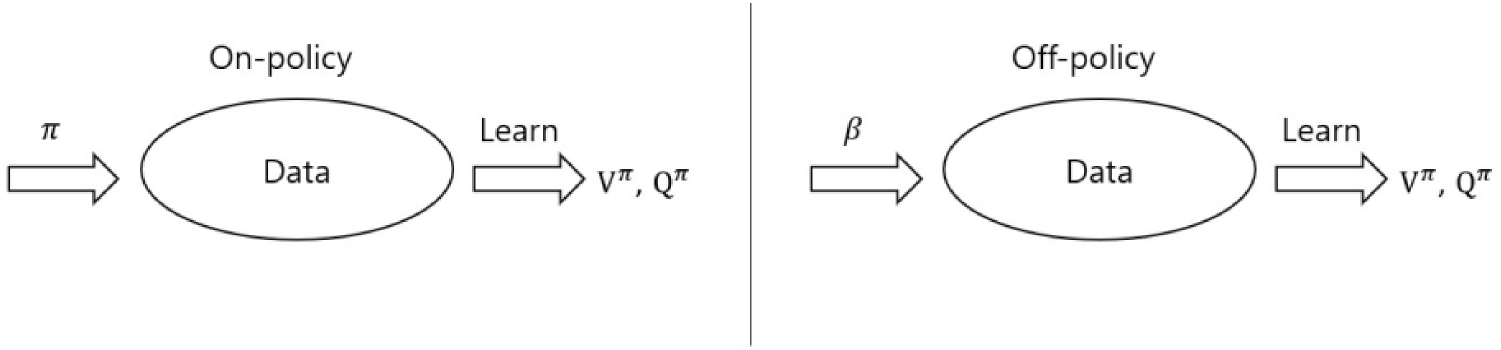


Figure 2.10: On/off policy learning

1. Target policy π: the policy that we want to learn, i.e., estimate V^π.
2. Behavior policy β: the policy that the agent follows to obtain the episode or trajectory.

1. On-policy learning: the target policy and behavior policies are identical ($\beta = \pi$), i.e., episodes are generated by following the target policy to learn the value of the target policy.
2. Off-policy learning: the target policy and behavior policies can be different ($\beta = \pi$ or $\beta \neq \pi$), i.e., episodes are generated by following the behavior policy to learn the value of the target policy. Decoupling the target and behavior policies give us greater engineering benefits.

4. Off-Policy Learning

$$\prod_{k=0}^{\tau-1} \pi(a_k|s_k)P(s_{k+1}|s_k,a_k)$$

$$G_t^{\pi/\mu} = \frac{\pi(A_t|S_t)}{\mu(A_t|S_t)} \cdot \frac{\pi(A_{t+1}|S_{t+1})}{\mu(A_{t+1}|S_{t+1})} \cdot \frac{\pi(A_{t+2}|S_{t+2})}{\mu(A_{t+2}|S_{t+2})} \cdots \frac{\pi(A_{T-1}|S_{T-1})}{\mu(A_{T-1}|S_{T-1})} G_t$$

$$= \left(\prod_{k=t}^{T-1} \frac{\pi(A_t|S_t)}{\mu(A_t|S_t)} \right) \cdot G_t \qquad s.t.\ \mu = 0 \rightarrow \pi = 0$$

The value function V^π is then expressed as

$$V^\pi(s) = E\left[G_0 |\, s_0 = s, \pi\right]$$

$$= E\left[\sum_{s_0=s, a_0\in A} \sum_{s_1\in A, a_1\in A} \cdots \sum_{s_\tau\in S} \left\{ \prod_{k=0}^{\tau-1} \pi(a_k|s_k)P(s_{k+1}|s_k,a_k) \right\} \sum_{i=0}^{\tau-1} \gamma^i R(s_i,a_i) \right]$$

$$= E\left[\sum_{s_0=s, a_0\in A} \sum_{s_1\in A, a_1\in A} \cdots \sum_{s_\tau\in S} \left\{ \prod_{k=0}^{\tau-1} \frac{\pi(a_k|s_k)}{\beta(a_k|s_k)} \beta(a_k|s_k)P(s_{k+1}|s_k,a_k) \right\} \sum_{i=0}^{\tau-1} \gamma^i R(s_i,a_i) \right]$$

$$= E\left[\sum_{s_0=s, a_0\in A} \sum_{s_1\in A, a_1\in A} \cdots \sum_{s_\tau\in S} \left\{ \prod_{k=0}^{\tau-1} \beta(a_k|s_k)P(s_{k+1}|s_k,a_k) \right\} \left\{ \prod_{j=0}^{\tau-1} \frac{\pi(a_j|s_j)}{\beta(a_j|s_j)} \right\} \sum_{i=0}^{\tau-1} \gamma^i R(s_i,a_i) \right]$$

$$= E\left[\left\{ \prod_{j=0}^{\tau-1} \frac{\pi(a_j|s_j)}{\beta(a_j|s_j)} \right\} G_0 \middle|\, s_0 = s, \beta \right]$$

4. Off-Policy Learning

Algorithm 7 Off-policy first-visit Monte Calro prediction (batch version)

Input: a policy π to be evaluated and a behavior policy β
Initialize
$V(s) = 0$ for all $s \in S$
$Returns(s) \leftarrow$ an empty list for all $s \in S$.
for $i \in \{0, 1, \ldots\}$ **do**
    Generate an episode following β: $(s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_{\tau-1}, a_{\tau-1}, r_{\tau-1}, s_\tau)$
    $G \leftarrow 0$
    for $k = \tau - 1, \tau - 2, \ldots, 0$ **do**
        $G \leftarrow \gamma G + r_k$
        if s_k does not appear in $s_0, s_1, \ldots, s_{k-1}$ **then**
            Append $\prod_{j=k}^{\tau-1} \frac{\pi(a_j|s_j)}{\beta(a_j|s_j)} G$ to the list $Rerturns(s_k)$
            $V(s_k) \leftarrow \text{average}(Returns(s_k))$
        end if
    end for
end for

Reference

[1] lecture2
[2] https://en.wikipedia.org/wiki/Law_of_large_numbers
[3] https://deeesp.github.io/statistics/Unbiased-Estimator/
[4] deepmind.com/learning-resources/-introduction-reinforcement-learning-david-silver
[5] https://sumniya.tistory.com/15
[6] https://analysisbugs.tistory.com/115
[7] https://data-newbie.tistory.com/534