MLM: Lecture 2

(Model-Free Policy Evaluation in RL + Intro to Model-Free Control)

Monday, February 27, 2023

(Heavily inspired by the Stanford RL Course of Prof. Emma Brunskill, but all potential errors are mine.)

Plan for The First Part

- Policy evaluation when we do not know the model (neither the state-transition probabilities, nor the reward functions).
- Two kinds of methods today (there are more out there):
  - Monte-Carlo Policy Evaluation
  - Temporal-Difference Learning

Plan for The First Part

- Policy evaluation when we do not know the model (neither the state-transition probabilities, nor the reward functions).

- Two kinds of methods today (there are more out there):
  - Monte-Carlo Policy Evaluation
  - Temporal-Difference Learning

Part 0: Reminder from Last Lecture

Markov Reward Process

Markov reward process = Markov process + Reward

Formally, MRP is given by:

- A set of states S.
- A transition model $P[X_{t+1} = s' \,|\, X_t = s]$, which we also denote by $P(s' \,|\, s)$.
- A reward function $R(s) = \mathbb{E}[R_t \,|\, X_t = s]$, which is the expected reward the agent receives in state s, $(s \in S)$.
- A discount factor $\gamma \in [0; 1]$.

Markov Reward Process

Markov reward process = Markov process + Reward

Formally, MRP is given by:

- A set of states S.
- A transition model $P[X_{t+1} = s' \,|\, X_t = s]$, which we also denote by $P(s' \,|\, s)$.
- A reward function $R(s) = \mathbb{E}[R_t \,|\, X_t = s]$, which is the expected reward the agent receives in state s, $(s \in S)$.
- A discount factor $\gamma \in [0; 1]$.

Markov Reward Process

Markov reward process = Markov process + Reward

Formally, MRP is given by:

- A set of states S.
- A transition model $P[X_{t+1} = s' \,|\, X_t = s]$, which we also denote by $P(s' \,|\, s)$.
- A reward function $R(s) = \mathbb{E}[R_t \,|\, X_t = s]$, which is the expected reward the agent receives in state s, $(s \in S)$.
- A discount factor $\gamma \in [0; 1]$.

Markov Reward Process

Markov reward process = Markov process + Reward

Formally, MRP is given by:

- A set of states S.
- A transition model $P[X_{t+1} = s' \,|\, X_t = s]$, which we also denote by $P(s' \,|\, s)$.
- A reward function $R(s) = \mathbb{E}[R_t \,|\, X_t = s]$, which is the expected reward the agent receives in state s, $(s \in S)$.
- A discount factor $\gamma \in [0; 1]$.

Markov Reward Process

Markov reward process = Markov process + Reward

Formally, MRP is given by:

- A set of states S.
- A transition model $P[X_{t+1} = s' \,|\, X_t = s]$, which we also denote by $P(s' \,|\, s)$.
- A reward function $R(s) = \mathbb{E}[R_t \,|\, X_t = s]$, which is the expected reward the agent receives in state s, $(s \in S)$.
- A discount factor $\gamma \in [0; 1]$.

Markov Reward Process

Markov reward process = Markov process + Reward

Formally, MRP is given by:

- A set of states S.
- A transition model $P[X_{t+1} = s' \,|\, X_t = s]$, which we also denote by $P(s' \,|\, s)$.
- A reward function $R(s) = \mathbb{E}[R_t \,|\, X_t = s]$, which is the expected reward the agent receives in state s, $(s \in S)$.
- A discount factor $\gamma \in [0; 1]$.

Return from an Episode

- **Horizon:**
  - Number of time steps in an episode (which can also be infinite). **We will first assume infinite horizons** (they are easier because they will lead to stationary, i.e. time-independent, policies!).

- **Return g_t:**
  - **Given:** An episode $s_1, s_2, s_3, s_4, \ldots, s_H$.
  - **Compute:** Return g_t = discounted sum of rewards from time t.
  - **As a formula:**

$$g_t = R(s_t) + R(s_{t+1}) \cdot \gamma + R(s_{t+2}) \cdot \gamma^2 + \ldots = R(s_t) + \sum_{i=1} R(s_{t+i}) \cdot \gamma^i$$

Return from an Episode

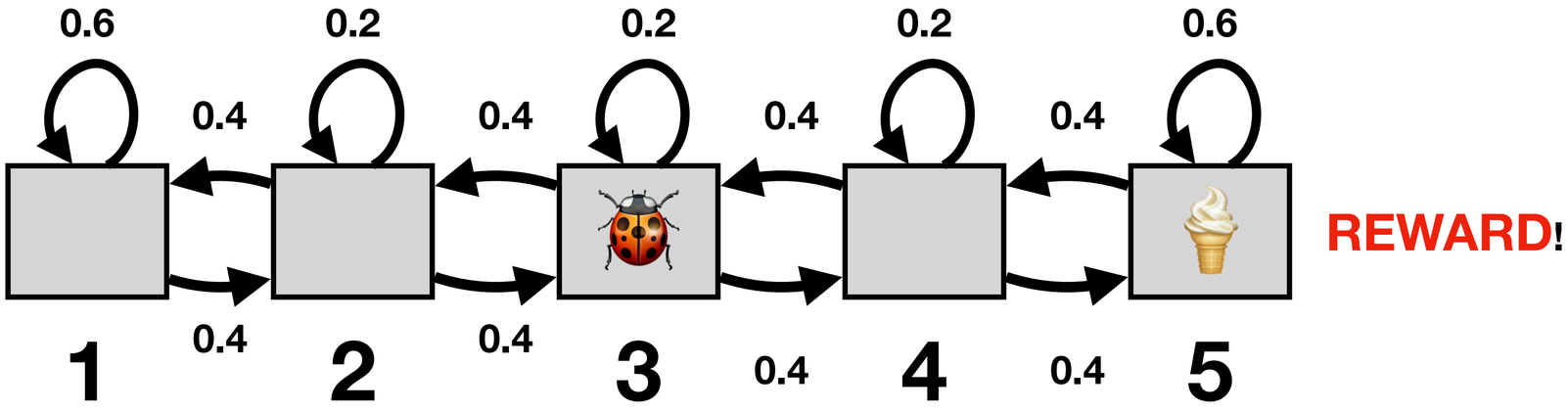
- **Horizon:**
  - Number of time steps in an episode (which can also be infinite). **We will first assume infinite horizons** (they are easier because they will lead to stationary, i.e. time-independent, policies!).

- **Return g_t:**
  - **Given:** An episode $s_1, s_2, s_3, s_4, \ldots, s_H$.
  - **Compute:** Return g_t = discounted sum of rewards from time t.
  - **As a formula:**

$$g_t = R(s_t) + R(s_{t+1}) \cdot \gamma + R(s_{t+2}) \cdot \gamma^2 + \ldots = R(s_t) + \sum_{i=1} R(s_{t+i}) \cdot \gamma^i$$

Markov Reward Process

Markov reward process = Markov process + Reward



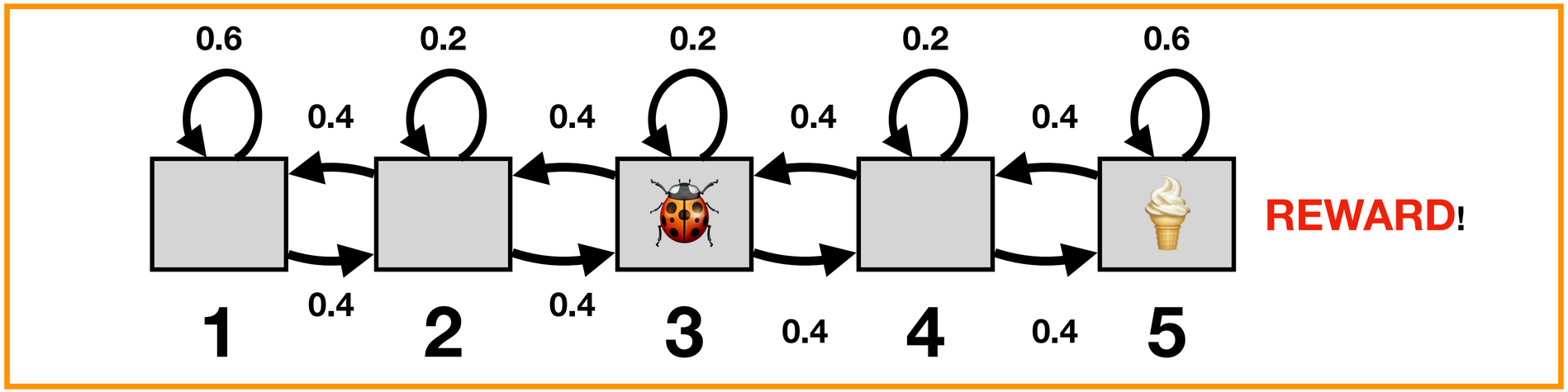
For example:

$$R(s) = \begin{cases} 0, & s = 1 \\ 0, & s = 2 \\ 0, & s = 3 \\ 0, & s = 4 \\ 10, & s = 5 \end{cases}$$

We expect that each time we visit s_5, there will be ice cream (i.e. we are not running out of it).

Markov Reward Process

Markov reward process = Markov process + Reward



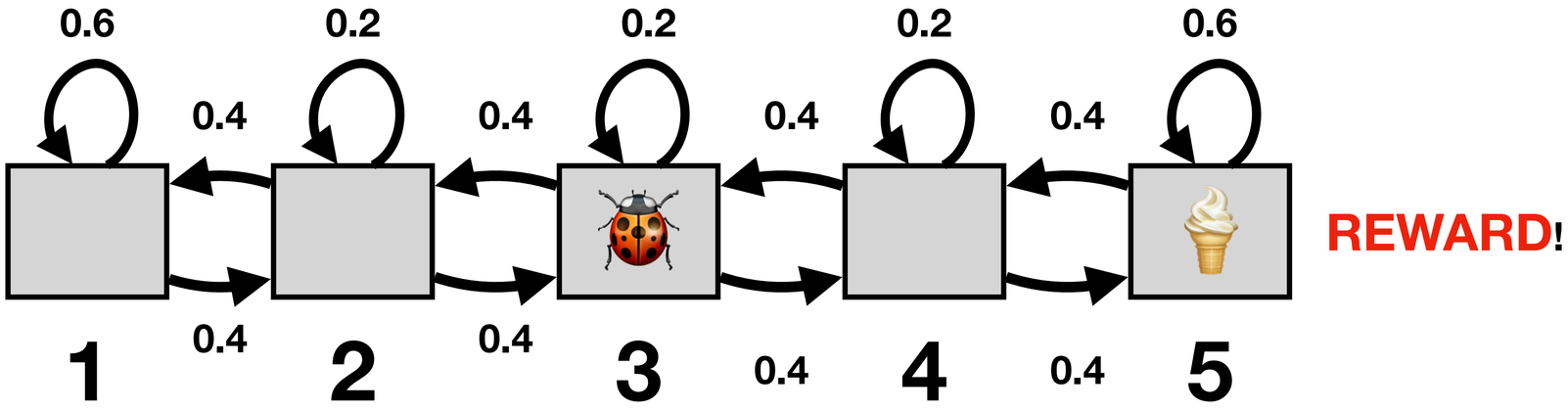
For example:

$$R(s) = \begin{cases} 0, & s = 1 \\ 0, & s = 2 \\ 0, & s = 3 \\ 0, & s = 4 \\ 10, & s = 5 \end{cases} \longleftarrow$$

We expect that each time we visit s_5, there will be ice cream (i.e. we are not running out of it).

Markov Reward Process

Markov reward process = Markov process + Reward

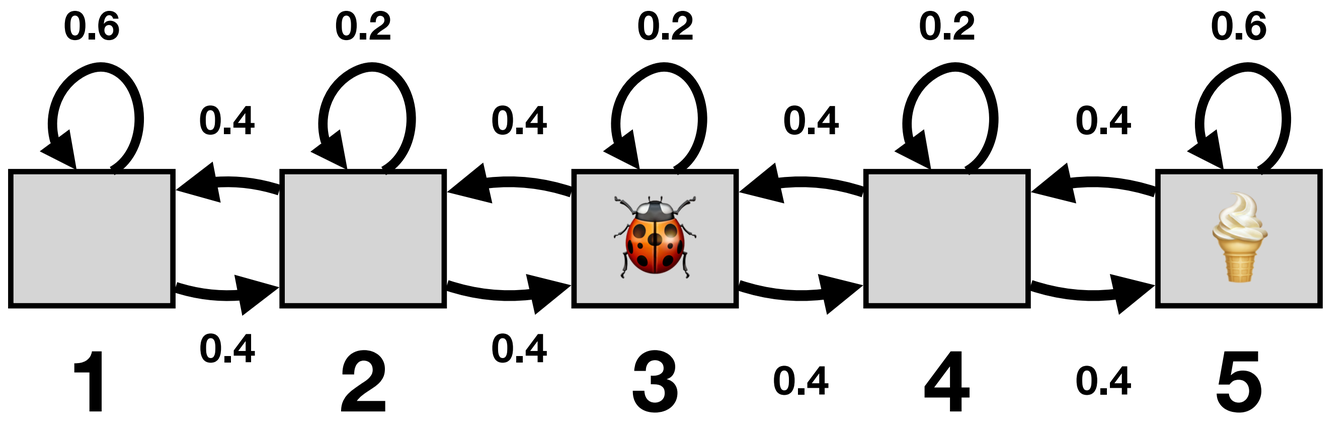


For example:

$$R(s) = \begin{cases} 0, & s = 1 \\ 0, & s = 2 \\ 0, & s = 3 \\ 0, & s = 4 \\ 10, & s = 5 \end{cases}$$

We expect that each time we visit s_5, there will be ice cream (i.e. we are not running out of it).

Episode (An Example)

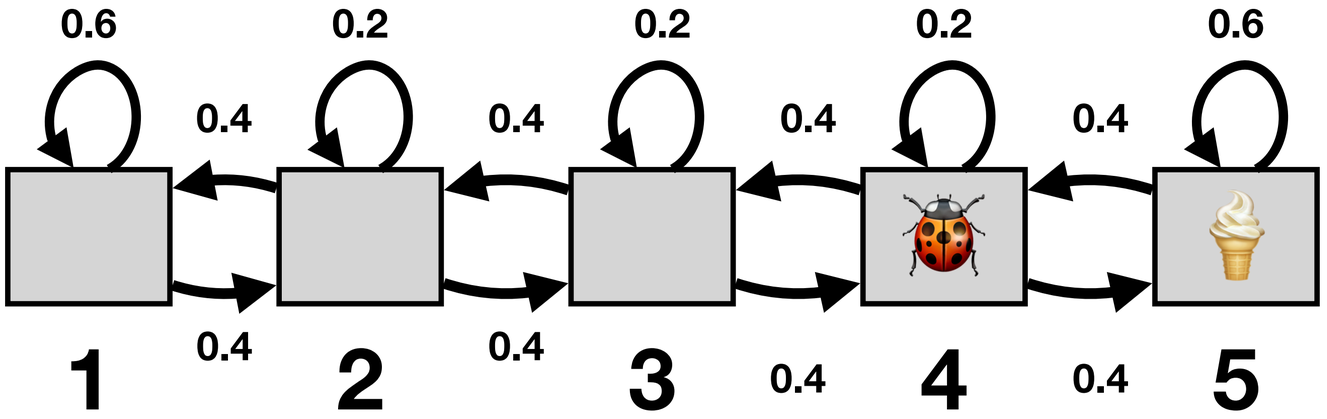


Time: $t = 1$

Current state: $s_1 = 3$, Current reward: $r_1 = 0$

Episode: 3

Episode (An Example)

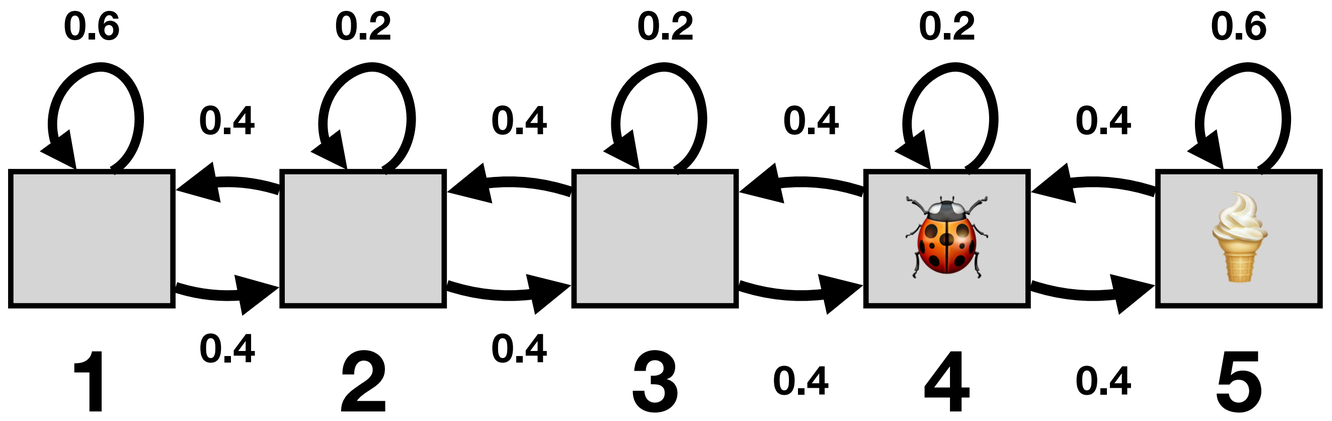


Time: $t = 2$

Current state: $s_2 = 4$, Current reward: $r_2 = 0$

Episode: $3, 4$

Episode (An Example)

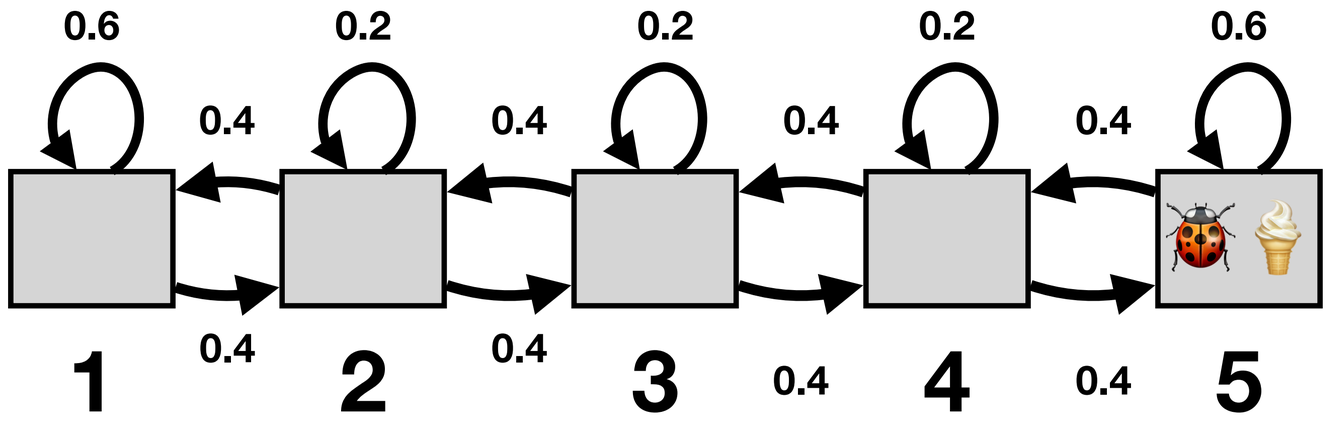


Time: $t = 3$

Current state: $s_3 = 4$, Current reward: $r_3 = 0$

Episode: $3, 4, 4$

Episode (An Example)

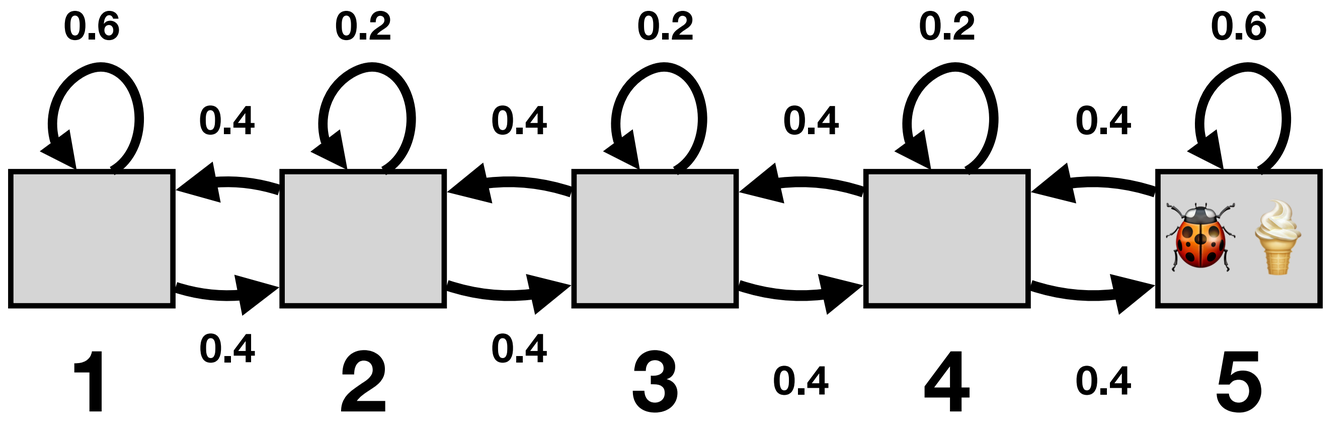


Time: $t = 4$

Current state: $s_4 = 5$, Current reward: $r_4 = 10$

Episode: $3, 4, 4, 5$

Episode (An Example)

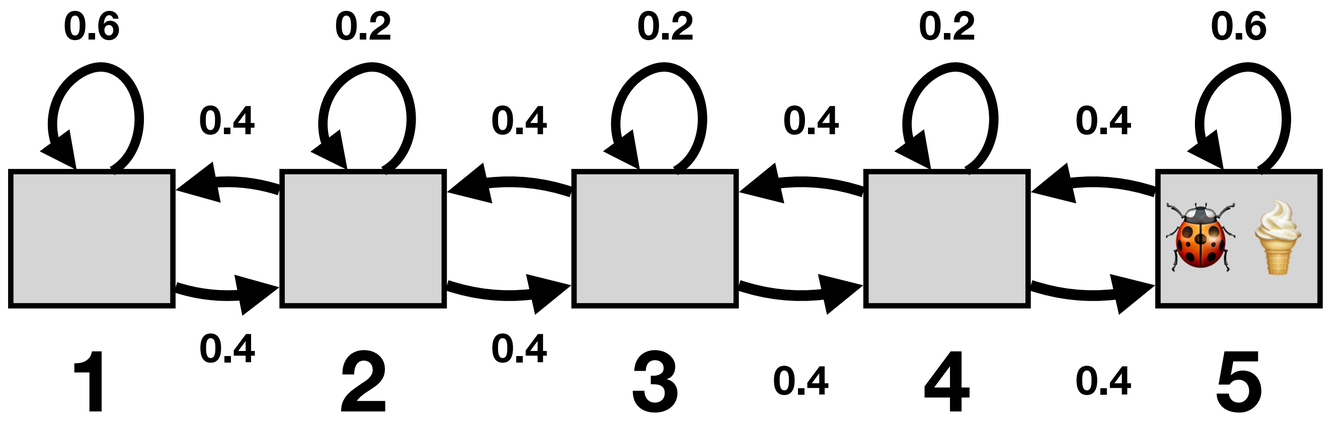


Time: $t = 5$

Current state: $s_4 = 5$, Current reward: $r_5 = 10$

Episode: $3, 4, 4, 5, 5$

Episode (An Example)



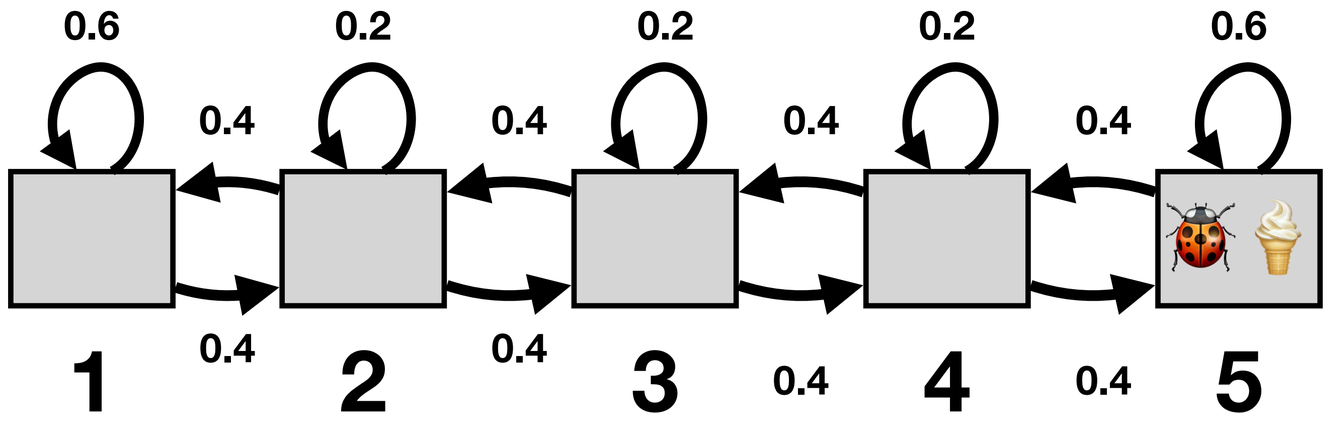
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + 0 \cdot 0.5 + 0 \cdot 0.5^2 + 10 \cdot 0.5^3 + 10 \cdot 0.5^4 = 1.875$$

Episode (An Example)



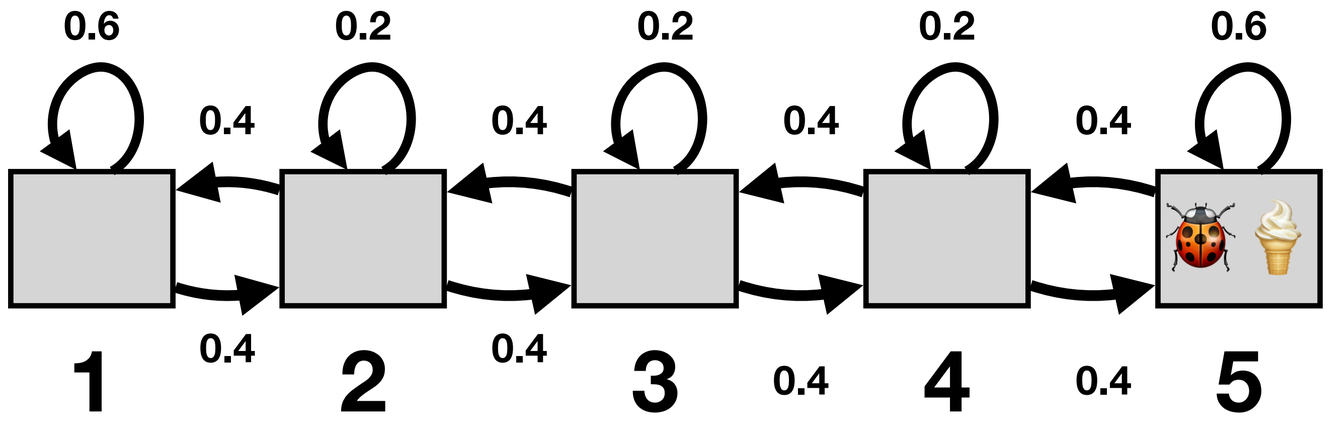
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + 0 \cdot 0.5 + 0 \cdot 0.5^2 + 10 \cdot 0.5^3 + 10 \cdot 0.5^4 = 1.875$$

Episode (An Example)



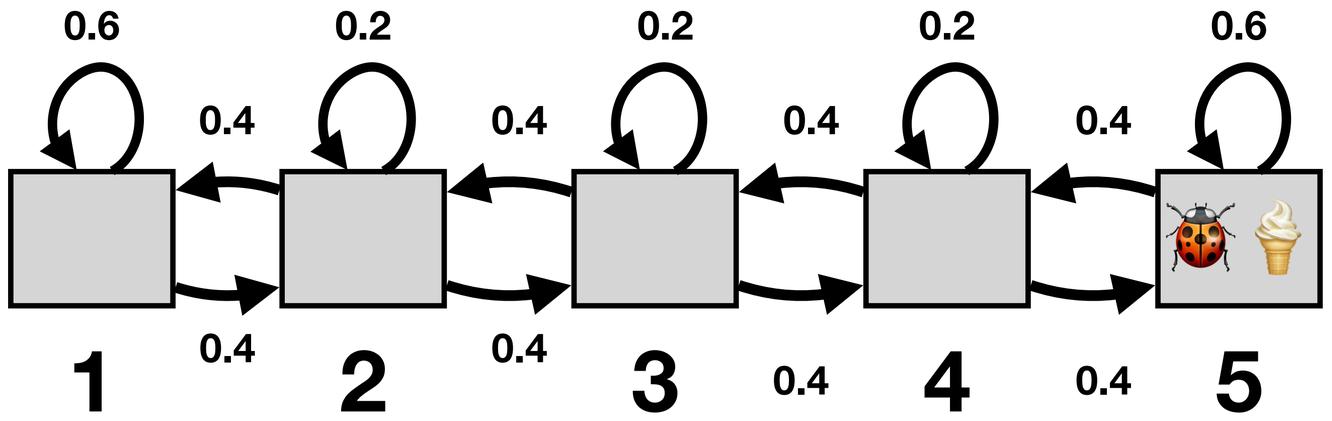
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + \boxed{0 \cdot 0.5} + 0 \cdot 0.5^2 + 10 \cdot 0.5^3 + 10 \cdot 0.5^4 = 1.875$$

Episode (An Example)



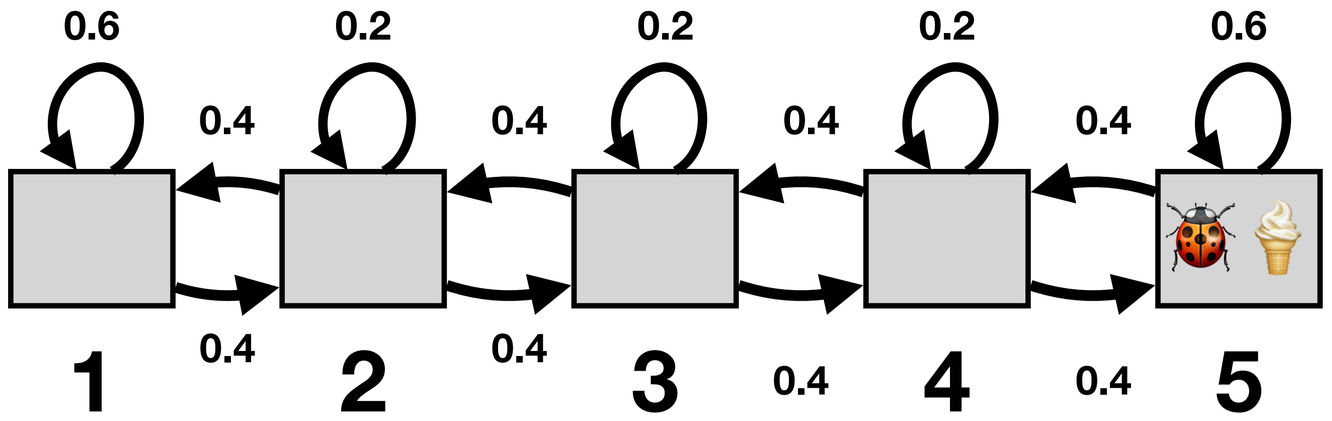
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + 0 \cdot 0.5 + 0 \cdot 0.5^2 + 10 \cdot 0.5^3 + 10 \cdot 0.5^4 = 1.875$$

Episode (An Example)



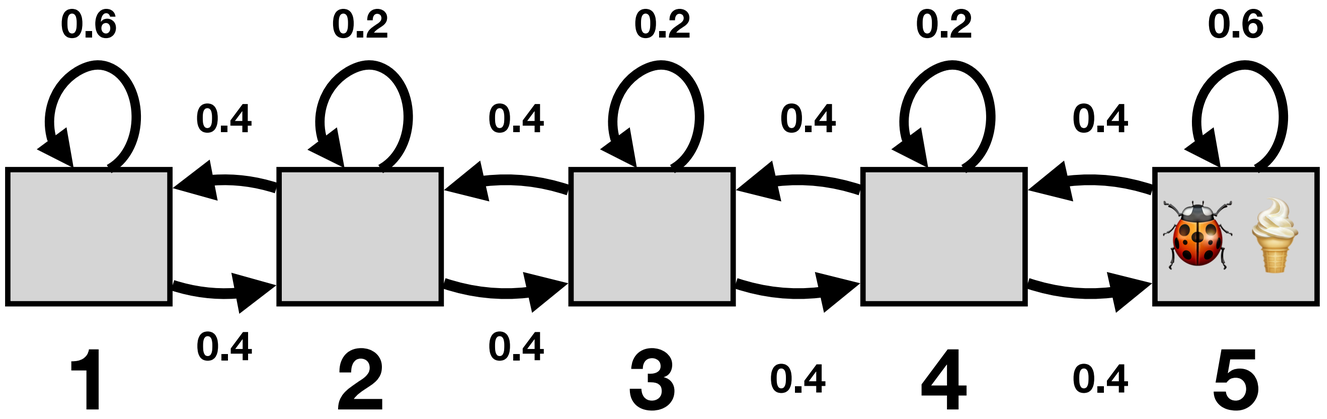
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + 0 \cdot 0.5 + 0 \cdot 0.5^2 + \boxed{10 \cdot 0.5^3} + 10 \cdot 0.5^4 = 1.875$$

Episode (An Example)



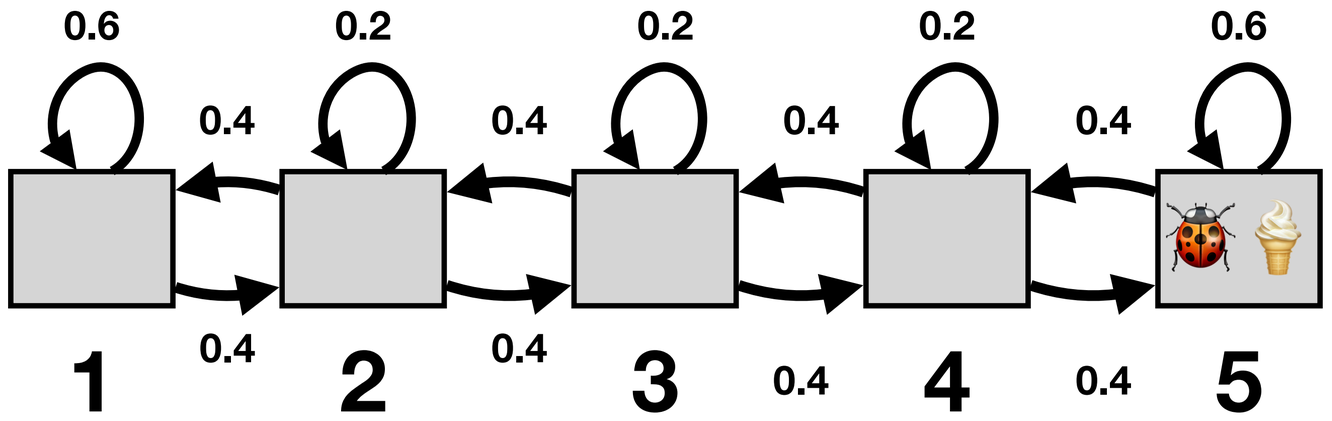
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + 0 \cdot 0.5 + 0 \cdot 0.5^2 + 10 \cdot 0.5^3 + \boxed{10 \cdot 0.5^4} = 1.875$$

Episode (An Example)



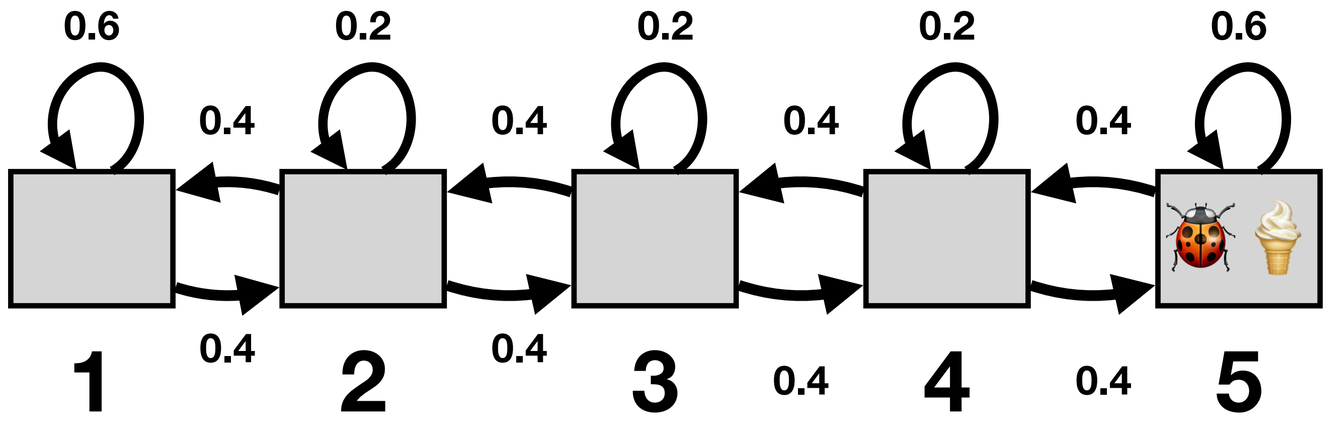
Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$$g_1 = 0 + 0 \cdot 0.5 + 0 \cdot 0.5^2 + 10 \cdot 0.5^3 + 10 \cdot 0.5^4 = 1.875$$

Episode (An Example)



Time: $t = 5$

Current state: $s_4 = 5$

Episode: $3, 4, 4, 5, 5$

$g_3 = 0 + 10 \cdot 0.5 + 10 \cdot 0.5^2 = 7.5$

Return (Random Variable)

- What we had on the previous slide was return from one specific sampled episode.

- Next we define **return** of a Markov reward process as a random variable (it is important to understand the distinction between the two):

$$G_t = R(X_t) + \gamma \cdot R(X_{t+1}) + \gamma^2 \cdot R(X_{t+2}) + \ldots = \sum_{i=0}^{\infty} R(X_{t+i}) \cdot \gamma^i$$

Return (Random Variable)

- What we had on the previous slide was return from one specific sampled episode.

- Next we define **return** of a Markov reward process as a random variable (it is important to understand the distinction between the two):

$$G_t = R(X_t) + \gamma \cdot R(X_{t+1}) + \gamma^2 \cdot R(X_{t+2}) + \ldots = \sum_{i=0}^{\infty} R(X_{t+i}) \cdot \gamma^i$$

Markov Decision Process

- **Markov decision process = Markov reward process + Actions**
- **An MDP is given by:**
  - A set of states S.
  - A set of actions A.
  - A transition model $P[X_{t+1} = s' \,|\, X_t = s, A_t = a] \underbrace{= P(s' \,|\, s, a)}_{\text{notation}}$
  - A reward $R(s, a) = \mathbb{E}[R_t \,|\, X_t = s, A_t = a]$, i.e. the expected reward that the agent receives when performing action a in state s.
  - Discount factor γ.

Markov Decision Process

- **Markov decision process = Markov reward process + Actions**
- **An MDP is given by:**
  - A set of states S.
  - A set of actions A.
  - A transition model $P[X_{t+1} = s' \,|\, X_t = s, A_t = a] = \underbrace{P(s' \,|\, s, a)}_{\text{notation}}$
  - A reward $R(s, a) = \mathbb{E}[R_t \,|\, X_t = s, A_t = a]$, i.e. the expected reward that the agent receives when performing action a in state s.
  - Discount factor γ.

Markov Decision Process

- **Markov decision process = Markov reward process + Actions**
- **An MDP is given by:**
  - A set of states S.
  - A set of actions A.
  - A transition model $P[X_{t+1} = s' \,|\, X_t = s, A_t = a] = \underbrace{P(s' \,|\, s, a)}_{\text{notation}}$
  - A reward $R(s, a) = \mathbb{E}[R_t \,|\, X_t = s, A_t = a]$, i.e. the expected reward that the agent receives when performing action a in state s.
  - Discount factor γ.

Markov Decision Process

- **Markov decision process = Markov reward process + Actions**
- **An MDP is given by:**
  - A set of states S.
  - A set of actions A.
  - A transition model $P[X_{t+1} = s' \,|\, X_t = s, A_t = a] = \underbrace{P(s' \,|\, s, a)}_{\text{notation}}$
  - A reward $R(s, a) = \mathbb{E}[R_t \,|\, X_t = s, A_t = a]$, i.e. the expected reward that the agent receives when performing action a in state s.
  - Discount factor γ.

Markov Decision Process

- **Markov decision process = Markov reward process + Actions**
- **An MDP is given by:**
  - A set of states S.
  - A set of actions A.
  - A transition model $P[X_{t+1} = s' \,|\, X_t = s, A_t = a] = \underbrace{P(s' \,|\, s, a)}_{\text{notation}}$
  - A reward $R(s, a) = \mathbb{E}[R_t \,|\, X_t = s, A_t = a]$, i.e. the expected reward that the agent receives when performing action a in state s.
  - Discount factor γ.

Markov Decision Process

- **Markov decision process = Markov reward process + Actions**
- **An MDP is given by:**
  - A set of states S.
  - A set of actions A.
  - A transition model $P[X_{t+1} = s' \,|\, X_t = s, A_t = a] = \underbrace{P(s' \,|\, s, a)}_{\text{notation}}$
  - A reward $R(s, a) = \mathbb{E}[R_t \,|\, X_t = s, A_t = a]$, i.e. the expected reward that the agent receives when performing action a in state s.
  - Discount factor γ.

Policy

- Policy determines which action to take in each state s.
- It can be either deterministic or random — that is also why policy will not simply be a function from states to actions.
- **We define policy:** $\pi(a \mid s) = P(A_t = a \mid X_t = s)$.
- **Example** (policy for our ladybug 🐞):
  - $A = \{\text{left}, \text{right}\}$
  - $\pi(\text{left} \mid 1) = 0$, $\pi(\text{right} \mid 1) = 1$, $\pi(\text{left} \mid 2) = 0.5$, $\pi(\text{right} \mid 1) = 0.5$,…

Policy

- Policy determines which action to take in each state s.

- It can be either deterministic or random — that is also why policy will not simply be a function from states to actions.

- **We define policy:** $\pi(a \mid s) = P(A_t = a \mid X_t = s)$.

- **Example** (policy for our ladybug 🐞):
  - $A = \{\text{left}, \text{right}\}$
  - $\pi(\text{left} \mid 1) = 0$, $\pi(\text{right} \mid 1) = 1$, $\pi(\text{left} \mid 2) = 0.5$, $\pi(\text{right} \mid 1) = 0.5, \ldots$

MDP+Policy = MRP

- When we specify a policy for a given MDP, we are effectively turning the MDP into a corresponding MRP.
- **Formally:**
  - Given an MDP (A, S, P, R, γ), we turn it into an MRP $(S, P^\pi, R^\pi, \gamma)$ where

$$P^\pi(s' \mid s) = \sum_{a \in A} \pi(a \mid s) \cdot P(s' \mid s, a) \text{ *}$$

$$R^\pi(s) = \sum_{a \in A} \pi(a \mid s) \cdot R(s, a)$$

* In the more verbose notation: $P^\pi[X_{t+1} = s' \mid X_t = s] = \sum_{a \in A} \pi(a \mid s) \cdot P[X_{t+1} = s' \mid A_t = a, X_t = s]$.

MDP+Policy = MRP

- When we specify a policy for a given MDP, we are effectively turning the MDP into a corresponding MRP.

- **Formally:**

  - Given an MDP (A, S, P, R, γ), we turn it into an MRP $(S, P^\pi, R^\pi, \gamma)$ where

$$P^\pi(s' \mid s) = \sum_{a \in A} \pi(a \mid s) \cdot P(s' \mid s, a) \text{ *}$$

$$R^\pi(s) = \sum_{a \in A} \pi(a \mid s) \cdot R(s, a)$$

* In the more verbose notation: $P^\pi[X_{t+1} = s' \mid X_t = s] = \sum_{a \in A} \pi(a \mid s) \cdot P[X_{t+1} = s' \mid A_t = a, X_t = s]$.

MDP+Policy = MRP

- When we specify a policy for a given MDP, we are effectively turning the MDP into a corresponding MRP.

- **Formally:**

  - Given an MDP (A, S, P, R, γ), we turn it into an MRP $(S, P^{\pi}, R^{\pi}, \gamma)$ where

$$P^{\pi}(s' \mid s) = \sum_{a \in A} \pi(a \mid s) \cdot P(s' \mid s, a) \text{ *}$$

$$R^{\pi}(s) = \sum_{a \in A} \pi(a \mid s) \cdot R(s, a)$$

* In the more verbose notation: $P^{\pi}[X_{t+1} = s' \mid X_t = s] = \sum_{a \in A} \pi(a \mid s) \cdot P[X_{t+1} = s' \mid A_t = a, X_t = s]$.

MDP+Policy = MRP

- When we specify a policy for a given MDP, we are effectively turning the MDP into a corresponding MRP.

- **Formally:**

  - Given an MDP (A, S, P, R, γ), we turn it into an MRP $(S, P^\pi, R^\pi, \gamma)$ where

$$P^\pi(s' \mid s) = \sum_{a \in A} \pi(a \mid s) \cdot P(s' \mid s, a) \ *$$

$$R^\pi(s) = \sum_{a \in A} \pi(a \mid s) \cdot R(s, a)$$

* In the more verbose notation: $P^\pi[X_{t+1} = s' \mid X_t = s] = \sum_{a \in A} \pi(a \mid s) \cdot P[X_{t+1} = s' \mid A_t = a, X_t = s]$.

MDP+Policy = MRP

- When we specify a policy for a given MDP, we are effectively turning the MDP into a corresponding MRP.

- **Formally:**

  - Given an MDP (A, S, P, R, γ), we turn it into an MRP $(S, P^{\pi}, R^{\pi}, \gamma)$ where

$$P^{\pi}(s' \mid s) = \sum_{a \in A} \pi(a \mid s) \cdot P(s' \mid s, a) \ *$$

$$R^{\pi}(s) = \sum_{a \in A} \pi(a \mid s) \cdot R(s, a)$$

* In the more verbose notation: $P^{\pi}[X_{t+1} = s' \mid X_t = s] = \sum_{a \in A} \pi(a \mid s) \cdot P[X_{t+1} = s' \mid A_t = a, X_t = s]$.

MDP+Policy = MRP

- When we specify a policy for a given MDP, we are effectively turning the MDP into a corresponding MRP.

- **Formally:**

  - Given an MDP (A, S, P, R, γ), we turn it into an MRP $(S, P^\pi, R^\pi, \gamma)$ where

$$P^\pi(s' \mid s) = \sum_{a \in A} \pi(a \mid s) \cdot P(s' \mid s, a) \ *$$

$$R^\pi(s) = \sum_{a \in A} \pi(a \mid s) \cdot R(s, a)$$

* In the more verbose notation: $P^\pi[X_{t+1} = s' \mid X_t = s] = \sum_{a \in A} \pi(a \mid s) \cdot P[X_{t+1} = s' \mid A_t = a, X_t = s]$.

MDP+Policy (An Example)

If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

The states:

MDP+Policy (An Example)

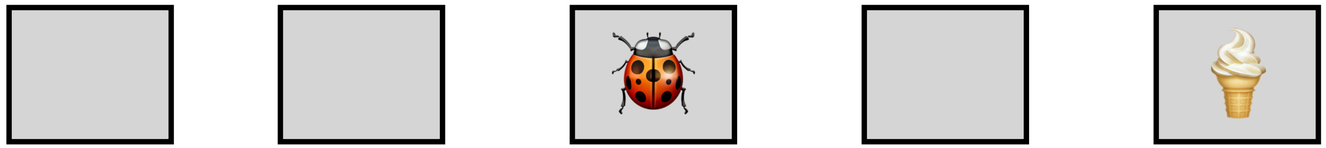
If we take the MDP with $S = \{1,2,3,4,5\}, A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

The states:



MDP+Policy (An Example)

If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

The states:

MDP+Policy (An Example)

If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

The states:

MDP+Policy (An Example)

If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

Now we will show the resulting Markov reward process:

MDP+Policy (An Example)

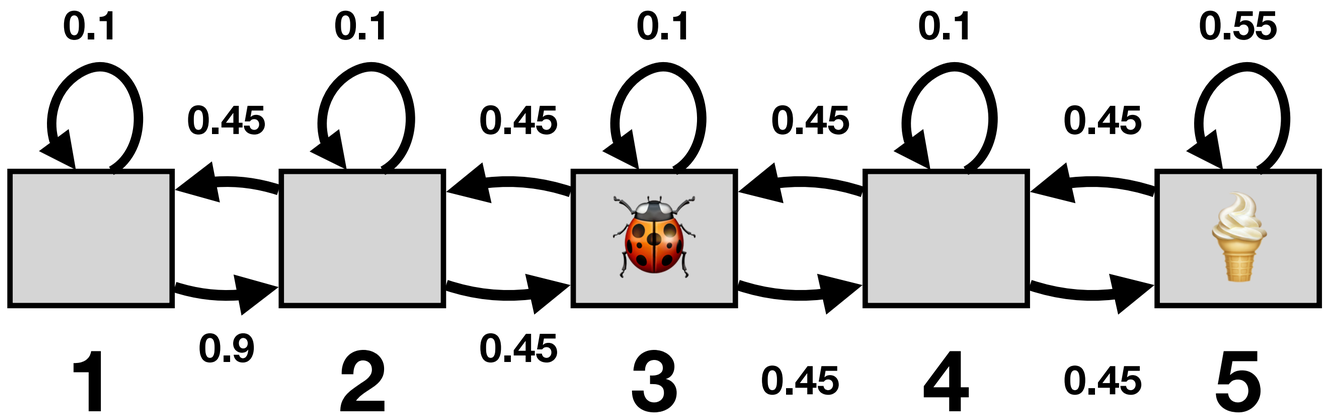
If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

…then we get the following Markov reward process:



MDP+Policy (An Example)

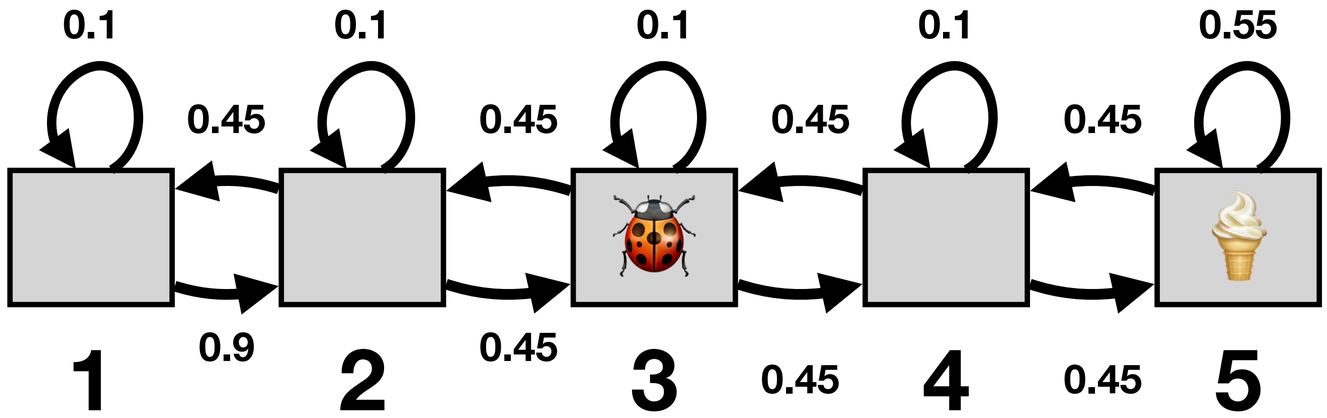
If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

…then we get the following Markov reward process:



For example:

$$P^{\pi}(2 \mid 3) = \pi(\textbf{left} \mid 3) \cdot P(2 \mid 3, \textbf{left}) +$$
$$+\pi(\textbf{right} \mid 3) \cdot P(2 \mid 3, \textbf{right}) +$$
$$+\pi(\textbf{eat} \mid 3) \cdot P(2 \mid 3, \textbf{eat}) =$$
$$= 0.5 \cdot 0.9 + 0.5 \cdot 0 + 0 \cdot 0 = 0.45$$

MDP+Policy (An Example)

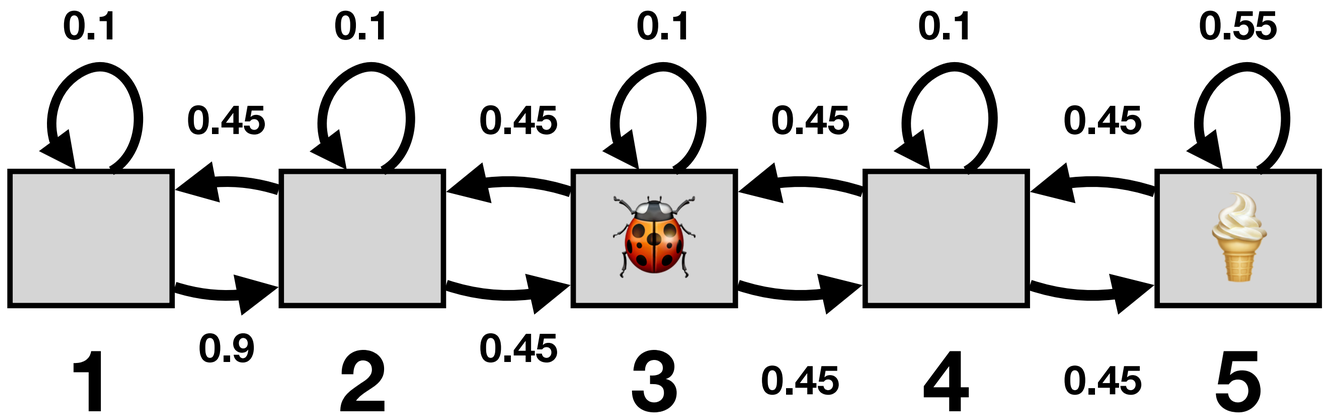
If we take the MDP with $S = \{1,2,3,4,5\}$, $A = \{\text{left}, \text{right}, \text{eat}\}$ and the state transition probabilties:

$$P(s' \mid s, \text{left}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s - s' = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{right}) = \begin{cases} 0.1 & s = s' \\ 0.9 & s' - s = 1, \\ 0 & \text{otherwise} \end{cases} \quad P(s' \mid s, \text{eat}) = \begin{cases} 1 & s = s' \\ 0 & \text{otherwise} \end{cases}$$

…and with the policy:

$$\pi(\text{left} \mid s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right} \mid s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat} \mid s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

…then we get the following Markov reward process:



For example:

$$P^{\pi}(2 \mid 2) = \pi(\textbf{left} \mid 2) \cdot P(2 \mid 2, \textbf{left}) +$$
$$+ \pi(\textbf{right} \mid 2) \cdot P(2 \mid 2, \textbf{right}) +$$
$$+ \pi(\textbf{eat} \mid 2) \cdot P(2 \mid 2, \textbf{eat}) =$$
$$= 0.5 \cdot 0.1 + 0.5 \cdot 0.1 + 0 \cdot 1 = 0.1$$

MDP+Policy (An Example)

Now, for the rewards, suppose the reward function of the MDP is:

$$R(s,a) = \begin{cases} 10 & s = 5 \text{ and } a = \text{eat} \\ 0 & \text{otherwise} \end{cases}$$

and we still use the same policy:

$$\pi(\text{left}\,|\,s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right}\,|\,s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat}\,|\,s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

then the reward function of the resulting Markov reward process is:

$$R^{\pi}(s) = \begin{cases} 5 & s = 5 \\ 0 & \text{otherwise} \end{cases},$$

MDP+Policy (An Example)

Now, for the rewards, suppose the reward function of the MDP is:

$$R(s,a) = \begin{cases} 10 & s = 5 \text{ and } a = \text{eat} \\ 0 & \text{otherwise} \end{cases}$$

and we still use the same policy:

$$\pi(\text{left}\,|\,s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right}\,|\,s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat}\,|\,s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

then the reward function of the resulting Markov reward process is:

$$R^{\pi}(s) = \begin{cases} 5 & s = 5 \\ 0 & \text{otherwise} \end{cases},$$

MDP+Policy (An Example)

Now, for the rewards, suppose the reward function of the MDP is:

$$R(s,a) = \begin{cases} 10 & s = 5 \text{ and } a = \text{eat} \\ 0 & \text{otherwise} \end{cases}$$

and we still use the same policy:

$$\pi(\text{left}\,|\,s) = \begin{cases} 0 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0.5 & s = 5 \end{cases} \quad \pi(\text{right}\,|\,s) = \begin{cases} 1 & s = 1 \\ 0.5 & s \in \{2,3,4\}, \\ 0 & s = 5 \end{cases} \quad \pi(\text{eat}\,|\,s) = \begin{cases} 0 & s \in \{1,2,3,4\} \\ 0.5 & s = 5 \end{cases}$$

then the reward function of the resulting Markov reward process is:

$$R^{\pi}(s) = \begin{cases} 5 & s = 5 \\ 0 & \text{otherwise} \end{cases},$$

here, for instance.

$$R^{\pi}(5) = \pi(\text{eat}\,|\,5) \cdot R(5,\text{eat}) + \pi(\text{left}\,|\,5) \cdot R(5,\text{left}) + \pi(\text{right}\,|\,5) \cdot R(5,\text{right}) = 0.5 \cdot 0 + 0.5 \cdot 10 + 0 \cdot 0 = 5$$

(State) Value Function

- **Definition:**

$$V(s) = \mathbb{E}[G_t \,|\, X_t = s] = \mathbb{E}[R(X_t) + \gamma \cdot R(X_{t+1}) + \gamma^2 \cdot R(X_{t+2}) + \ldots \,|\, X_t = s]$$

- **Intuition:** Value function $V(s)$ is the expected return when starting from state s.

(State) Value Function

- **Definition:**

$$V(s) = \mathbb{E}[G_t \,|\, X_t = s] = \mathbb{E}[R(X_t) + \gamma \cdot R(X_{t+1}) + \gamma^2 \cdot R(X_{t+2}) + \ldots \,|\, X_t = s]$$

- **Intuition:** Value function $V(s)$ is the expected return when starting from state s.

(State) Value Function

- **Definition:**

$$V(s) = \mathbb{E}[G_t \,|\, X_t = s] = \mathbb{E}[R(X_t) + \gamma \cdot R(X_{t+1}) + \gamma^2 \cdot R(X_{t+2}) + \ldots \,|\, X_t = s]$$

- **Intuition:** Value function $V(s)$ is the expected return when starting from state s.

(Bellman equation for MDP)

State Value Function of MDP

General case:

$$V^{\pi}(s) = \sum_{a \in A} \pi(a, s) \cdot \left[R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, a) \cdot V^{\pi}(s') \right]$$

Version for deterministic policy:

$$V^{\pi}(s) = R(s, \pi(s)) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, \pi(s)) \cdot V^{\pi}(s')$$

Part 1: Problem Statement

Problem: Model-Free Policy Evaluation

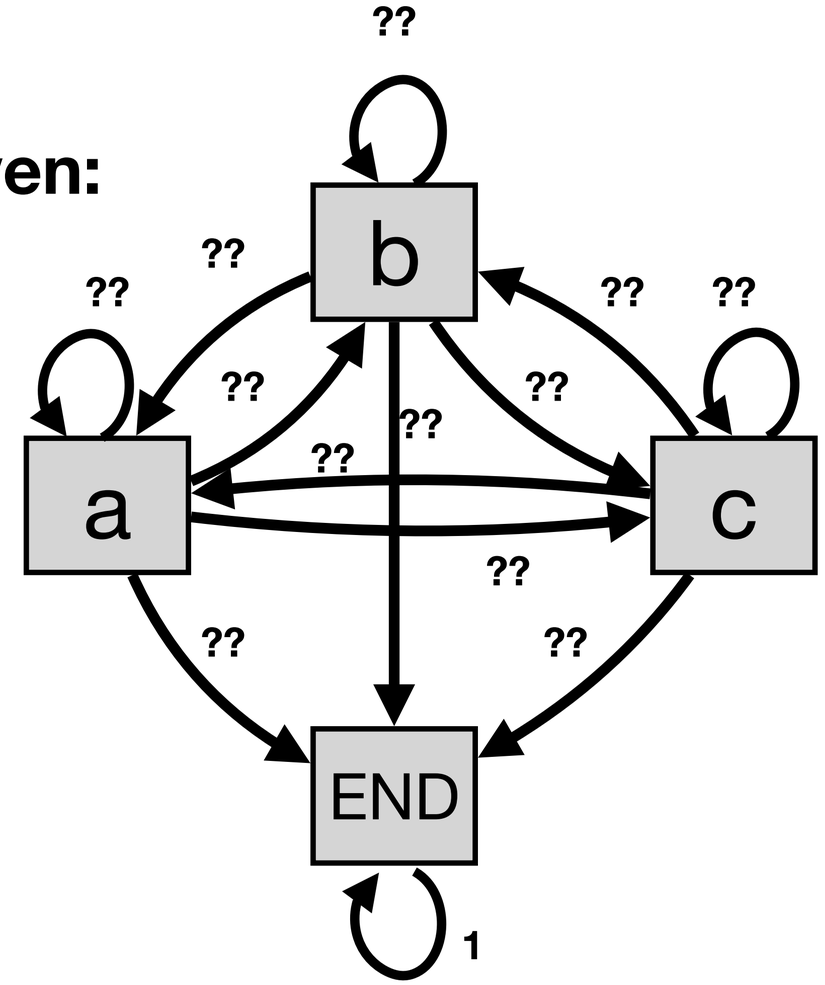
- Given a policy and an MDP with unknown parameters (or generally an environment with which we can interact), **estimate the value function.**

Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{l, r\}$$

Policy is given, e.g.:

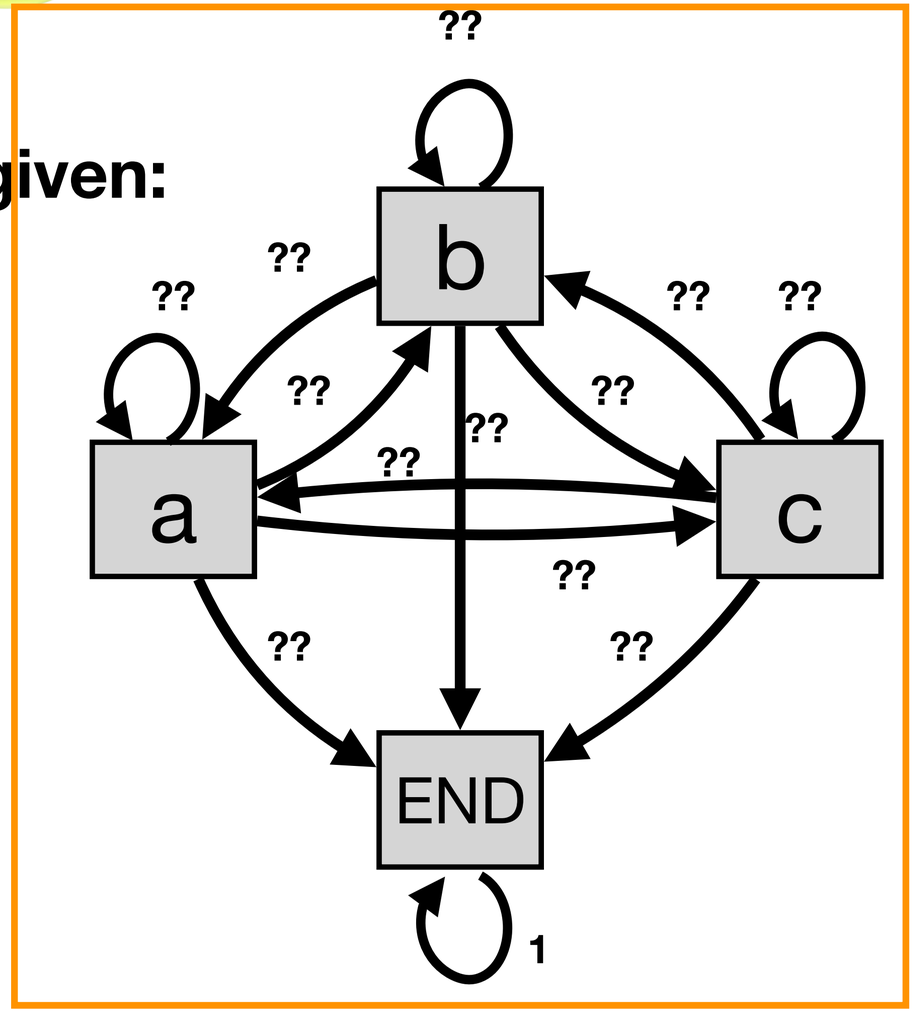
$$\pi(l \mid a) = 0.2,\ \pi(r \mid a) = 0.8,$$
$$\pi(l \mid b) = 0.3,\ \pi(r \mid b) = 0.7,$$
$$\ldots$$

Example

Agent:

Rewards??

States are given:



Actions are given:

$$A = \{l, r\}$$

Policy is given, e.g.:

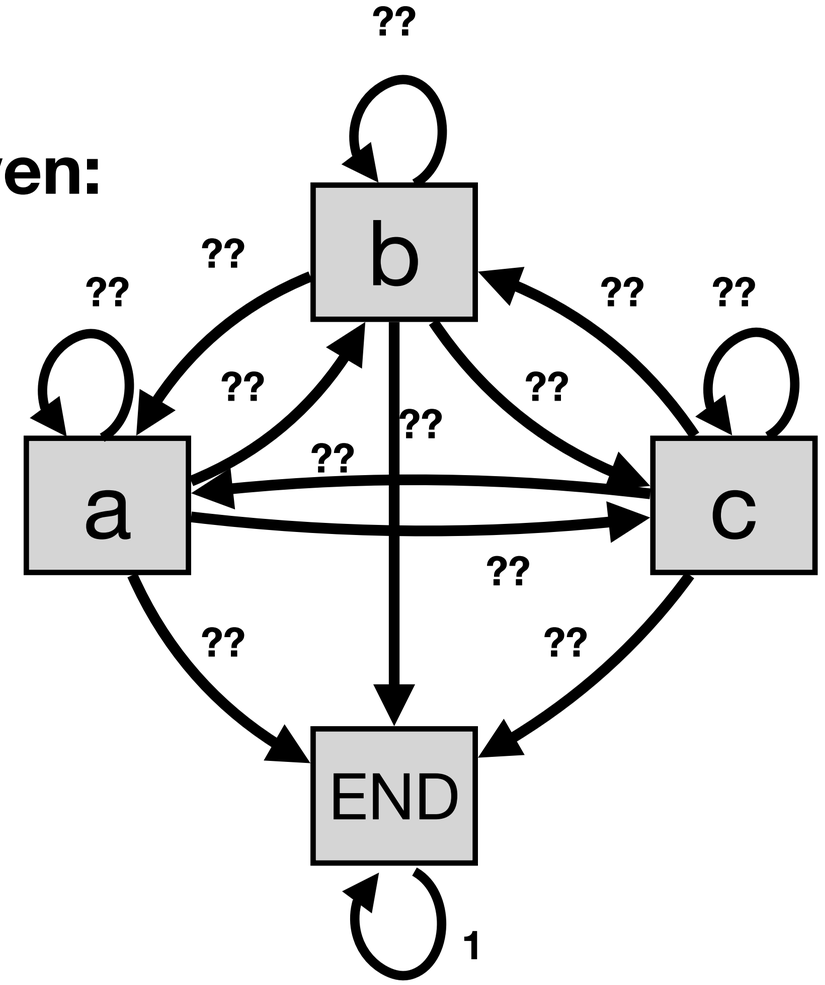
$$\pi(l \mid a) = 0.2,\ \pi(r \mid a) = 0.8,$$
$$\pi(l \mid b) = 0.3,\ \pi(r \mid b) = 0.7,$$
$$\ldots$$

Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{l, r\}$$

Policy is given, e.g.:

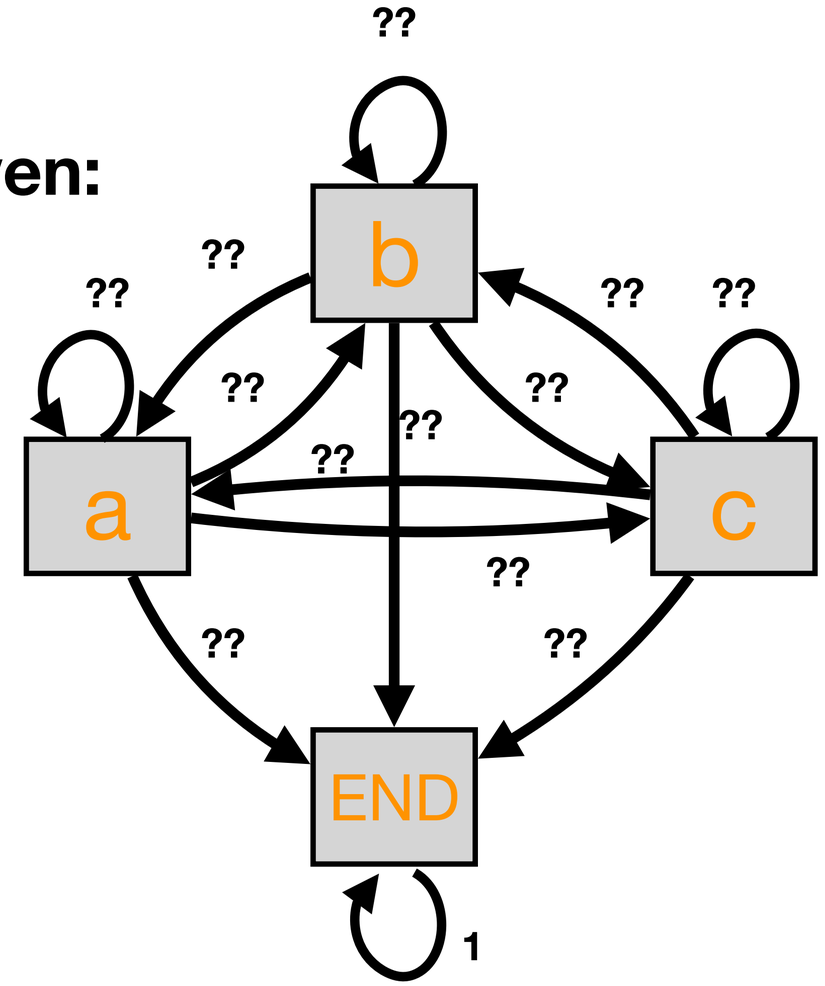
$$\pi(l \mid a) = 0.2,\ \pi(r \mid a) = 0.8,$$
$$\pi(l \mid b) = 0.3,\ \pi(r \mid b) = 0.7,$$
$$\dots$$

Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{l, r\}$$

Policy is given, e.g.:

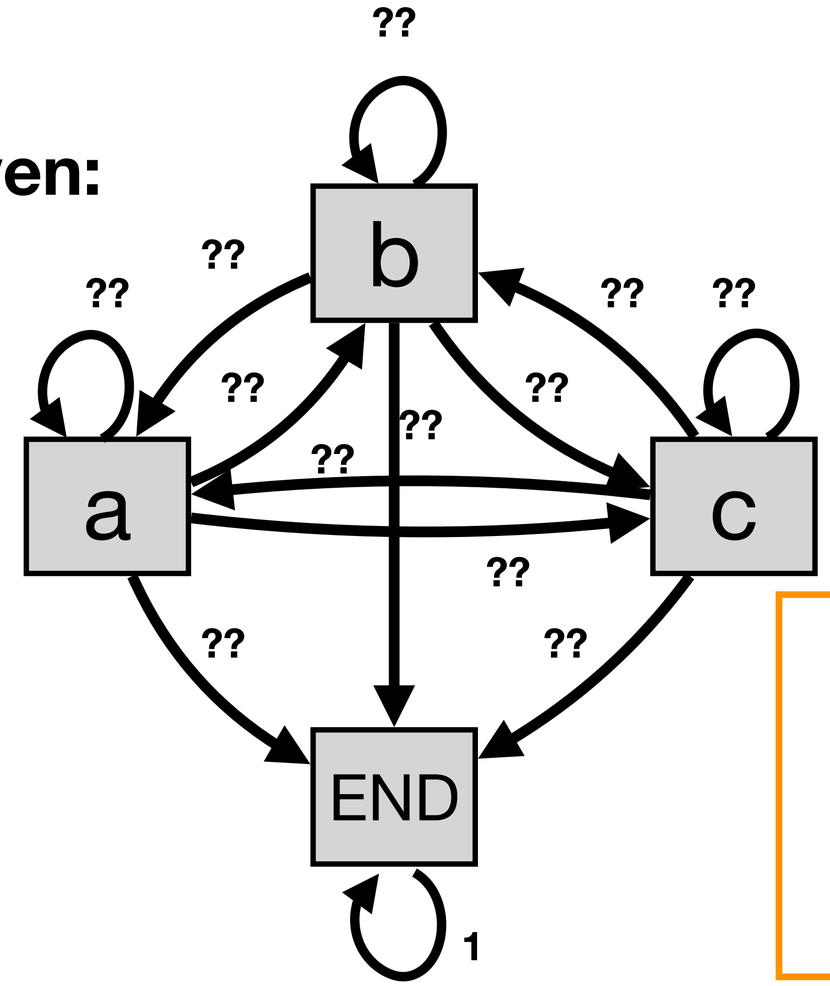
$$\pi(l \mid a) = 0.2,\ \pi(r \mid a) = 0.8,$$
$$\pi(l \mid b) = 0.3,\ \pi(r \mid b) = 0.7,$$
$$\ldots$$

Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{l, r\}$$

Policy is given, e.g.:

$$\pi(l \mid a) = 0.2,\ \pi(r \mid a) = 0.8,$$
$$\pi(l \mid b) = 0.3,\ \pi(r \mid b) = 0.7,$$
$$\ldots$$

Problem: Model-Free Policy Evaluation

- **Our task again:**
  - Given a policy and an MDP with unknown parameters (or generally an environment with which we can interact), **estimate the value function.**

An Assumption

- **Assumption:** In what follows we will assume that our MDP has terminal states and that the probability of infinitely long runs is zero.
- **Terminal states:** Once the system gets into a terminal state, it stays in it. The reward in the terminal state is always 0.
- **Why do we do this?** This assumption will allow us to use the formalism for infinite-horizon problems (which is mathematically simpler).

An Assumption

- **Assumption:** In what follows we will assume that our MDP has terminal states and that the probability of infinitely long runs is zero.

- **Terminal states:** Once the system gets into a terminal state, it stays in it. The reward in the terminal state is always 0.

- **Why do we do this?** This assumption will allow us to use the formalism for infinite-horizon problems (which is mathematically simpler).

An Assumption

- **Assumption:** In what follows we will assume that our MDP has terminal states and that the probability of infinitely long runs is zero.
- **Terminal states:** Once the system gets into a terminal state, it stays in it. The reward in the terminal state is always 0.
- **Why do we do this?** This assumption will allow us to use the formalism for infinite-horizon problems (which is mathematically simpler).

An Assumption

- **Assumption:** In what follows we will assume that our MDP has terminal states and that the probability of infinitely long runs is zero.

- **Terminal states:** Once the system gets into a terminal state, it stays in it. The reward in the terminal state is always 0.

- **Why do we do this?** This assumption will allow us to use the formalism for infinite-horizon problems (which is mathematically simpler).

Part 2: Statistical Properties of Estimators

(An informal recap of what you already know from statistics)

Estimators (Statistics)

- **Typical setting:**
  - We are given a sample of random variables $X_1, X_2, \ldots, X_n$.
  - Suppose that we want to estimate some parameter θ, e.g., suppose all the X_i's are sampled independently from the same distribution and we want to estimate the mean of this distribution.
  - An **estimator of** θ is a function $\hat{\theta}$ that maps samples to estimates of the parameter θ.

Estimators (Statistics)

- **Typical setting:**
  - We are given a sample of random variables $X_1, X_2, \ldots, X_n$.
  - Suppose that we want to estimate some parameter θ, e.g., suppose all the X_i's are sampled independently from the same distribution and we want to estimate the mean of this distribution.
  - An **estimator of** θ is a function $\hat{\theta}$ that maps samples to estimates of the parameter θ.

Estimators (Statistics)

- **Typical setting:**
  - We are given a sample of random variables $X_1, X_2, \ldots, X_n$.
  - Suppose that we want to estimate some parameter θ, e.g., suppose all the X_i's are sampled independently from the same distribution and we want to estimate the mean of this distribution.
  - An **estimator of** θ is a function $\hat{\theta}$ that maps samples to estimates of the parameter θ.

Estimators (Statistics)

- **Typical setting:**
  - We are given a sample of random variables $X_1, X_2, \ldots, X_n$.
  - Suppose that we want to estimate some parameter θ, e.g., suppose all the X_i's are sampled independently from the same distribution and we want to estimate the mean of this distribution.
  - An **estimator of** θ is a function $\hat{\theta}$ that maps samples to estimates of the parameter θ.

Estimators as Random Variables

- **Example:** Let us have a normal distribution with mean μ and standard deviation σ. Denote by $\mathbf{X} = (X_1, X_2, \ldots, X_N)$ an independent sample from this distribution. Then the sample mean $\hat{\mu}(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} X_i$ is an estimator for the population mean μ.
- Note that, in this example, $\hat{\mu}(\mathbf{X})$ is a **random variable**.

Estimators as Random Variables

- **Example:** Let us have a normal distribution with mean μ and standard deviation σ. Denote by $\mathbf{X} = (X_1, X_2, \ldots, X_N)$ an independent sample from this distribution. Then the sample mean $\hat{\mu}(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} X_i$ is an estimator for the population mean μ.

- Note that, in this example, $\hat{\mu}(\mathbf{X})$ is a **random variable**.

Estimators as Random Variables

- **Example:** Let us have a normal distribution with mean μ and standard deviation σ. Denote by $\mathbf{X} = (X_1, X_2, \ldots, X_N)$ an independent sample from this distribution. Then the sample mean $\hat{\mu}(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} X_i$ is an estimator for the population mean μ.

- Note that, in this example, $\hat{\mu}(\mathbf{X})$ is a **random variable**.

Estimators as Random Variables

- **Example:** Let us have a normal distribution with mean μ and standard deviation σ. Denote by $\mathbf{X} = (X_1, X_2, \ldots, X_N)$ an independent sample from this distribution. Then the sample mean $\hat{\mu}(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} X_i$ is an estimator for the population mean μ.

- Note that, in this example, $\hat{\mu}(\mathbf{X})$ is a **random variable**.

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.$

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.$

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.$

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.$

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.$

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.$

Bias

Bias of an estimator $\hat{\theta}$ is defined as: $\text{BIAS}_\theta(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}(\mathbf{X})] - \theta$.

If $\text{BIAS}_\theta(\hat{\theta}) = 0$ then we say that $\hat{\theta}$ is an unbiased estimator.

Example: $\frac{1}{N}\sum_{k=1}^{N} X_k$ is an unbiased estimator of population mean. Why?

Because we have $\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} X_k\right] = \frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[X_k\right] = \boxed{\frac{1}{N}\cdot N \cdot \mathbb{E}\left[X_k\right] = \mu.}$

Mean Squared Error

Mean squared error of an estimator $\hat{\theta}$ is defined as: $\mathrm{MSE}_\theta(\hat{\theta}) = \mathbb{E}_\theta[(\hat{\theta}(\mathbf{X}) - \theta)^2]$.

It holds $MSE_\theta(\hat{\theta}(\mathbf{X})) = \mathrm{Var}_\theta(\hat{\theta}(\mathbf{X})) + \mathrm{BIAS}(\hat{\theta}(\mathbf{X}))^2$.

Mean Squared Error

Mean squared error of an estimator $\hat{\theta}$ is defined as: $\mathrm{MSE}_\theta(\hat{\theta}) = \mathbb{E}_\theta[(\hat{\theta}(\mathbf{X}) - \theta)^2]$.

It holds $MSE_\theta(\hat{\theta}(\mathbf{X})) = \mathrm{Var}_\theta(\hat{\theta}(\mathbf{X})) + \mathrm{BIAS}(\hat{\theta}(\mathbf{X}))^2$.

Consistency

Let $\mathbf{X}_N = (X_1, \ldots, X_N)$ be an independent sample, used to estimate θ.

A sequence of estimators $\hat{\theta}_N(\mathbf{X}_N)$ is said to be consistent if for every $\varepsilon > 0$ it holds: $\lim_{N \to \infty} P[\,|\hat{\theta}_N(\mathbf{X}_N) - \theta| < \varepsilon] = 1.$

Consistency

Let $\mathbf{X}_N = (X_1, \ldots, X_N)$ be an independent sample, used to estimate θ.

A sequence of estimators $\hat{\theta}_N(\mathbf{X}_N)$ is said to be consistent if for every $\varepsilon > 0$ it holds: $\lim_{N \to \infty} P[\,|\hat{\theta}_N(\mathbf{X}_N) - \theta| < \varepsilon] = 1.$

Consistency

Let $\mathbf{X}_N = (X_1, \ldots, X_N)$ be an independent sample, used to estimate θ.

A sequence of estimators $\hat{\theta}_N(\mathbf{X}_N)$ is said to be consistent if for every $\varepsilon > 0$ it holds: $\lim_{N \to \infty} P[\,|\hat{\theta}_N(\mathbf{X}_N) - \theta| < \varepsilon] = 1.$

Why It Matters

- Estimators that we are going to study in this lecture can be analyzed in the same framework. After all, they are just statistical estimators.

Part 3: Monte-Carlo Policy Evaluation

Monte-Carlo Policy Evaluation (1/5)

Recall the definition of G_t, the return at time t *(we have not shown it explicitly for MDPs last time):*

$$G_t^\pi = R(X_t, A_t) + \gamma \cdot R(X_{t+1}, A_{t+1}) + \gamma^2 \cdot R(X_{t+2}, A_{t+2}) + \ldots = \sum_{i=0}^{\infty} R(X_{t+i}, A_{t+i}) \cdot \gamma^i$$

(for simplicity, we assume that the reward when R(a,s) is deterministic)

where X_i's and $A_i's$ are random variables — X_i is the state at time t and A_i is the action at time i. We suppose that these random variables are from an MDP with a policy π (which together define the distribution of these random variables).

Monte-Carlo Policy Evaluation (1/5)

Recall the definition of G_t, the return at time t *(we have not shown it explicitly for MDPs last time):*

$$G_t^\pi = R(X_t, A_t) + \gamma \cdot R(X_{t+1}, A_{t+1}) + \gamma^2 \cdot R(X_{t+2}, A_{t+2}) + \ldots = \sum_{i=0}^{\infty} R(X_{t+i}, A_{t+i}) \cdot \gamma^i$$

(for simplicity, we assume that the reward when R(a,s) is deterministic)

where X_i's and $A_i's$ are random variables — X_i is the state at time t and A_i is the action at time i. We suppose that these random variables are from an MDP with a policy π (which together define the distribution of these random variables).

Monte-Carlo Policy Evaluation (1/5)

Recall the definition of G_t, the return at time t *(we have not shown it explicitly for MDPs last time):*

$$G_t^\pi = R(X_t, A_t) + \gamma \cdot R(X_{t+1}, A_{t+1}) + \gamma^2 \cdot R(X_{t+2}, A_{t+2}) + \ldots = \sum_{i=0}^{\infty} R(X_{t+i}, A_{t+i}) \cdot \gamma^i$$

(for simplicity, we assume that the reward when R(a,s) is deterministic)

where X_i's and $A_i's$ are random variables — X_i is the state at time t and A_i is the action at time i. We suppose that these random variables are from an MDP with a policy π (which together define the distribution of these random variables).

Monte-Carlo Policy Evaluation (2/5)

The state value function $V^{\pi}(s)$ is*:*

$$V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \,|\, X_t = s].$$

We were computing $V^{\pi}(s)$ by solving the Bellman equation (directly or iteratively):

$$V^{\pi}(s) = \sum_{a \in A} \pi(a, s) \cdot \left[R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \,|\, s, a) \cdot V^{\pi}(s') \right].$$

But there is also another way to approximate $V^{\pi}(s)$. *

*T*his method will not be very efficient for MDPs but bear with me… we are getting somewhere)*

Monte-Carlo Policy Evaluation (2/5)

The state value function $V^{\pi}(s)$ is:

$$V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \mid X_t = s].$$

We were computing $V^{\pi}(s)$ by solving the Bellman equation (directly or iteratively):

$$V^{\pi}(s) = \sum_{a \in A} \pi(a, s) \cdot \left[R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, a) \cdot V^{\pi}(s') \right].$$

But there is also another way to approximate $V^{\pi}(s)$. *

*T*his method will not be very efficient for MDPs but bear with me… we are getting somewhere)*

Monte-Carlo Policy Evaluation (2/5)

The state value function $V^{\pi}(s)$ is:

$$V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \,|\, X_t = s].$$

We were computing $V^{\pi}(s)$ by solving the Bellman equation (directly or iteratively):

$$V^{\pi}(s) = \sum_{a \in A} \pi(a, s) \cdot \left[R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \,|\, s, a) \cdot V^{\pi}(s') \right].$$

But there is also another way to approximate $V^{\pi}(s)$. *

*T*his method will not be very efficient for MDPs but bear with me… we are getting somewhere)*

Monte-Carlo Policy Evaluation (2/5)

The state value function $V^{\pi}(s)$ is:

$$V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \mid X_t = s].$$

We were computing $V^{\pi}(s)$ by solving the Bellman equation (directly or iteratively):

$$V^{\pi}(s) = \sum_{a \in A} \pi(a, s) \cdot \left[R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, a) \cdot V^{\pi}(s') \right].$$

But there is also another way to approximate $V^{\pi}(s)$. *

*T*his method will not be very efficient for MDPs but bear with me… we are getting somewhere)*

Monte-Carlo Policy Evaluation (2/5)

The state value function $V^{\pi}(s)$ is:

$$V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \mid X_t = s].$$

We were computing $V^{\pi}(s)$ by solving the Bellman equation (directly or iteratively):

$$V^{\pi}(s) = \sum_{a \in A} \pi(a, s) \cdot \left[R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, a) \cdot V^{\pi}(s') \right].$$

But there is also another way to approximate $V^{\pi}(s)$. *

*T*his method will not be very efficient for MDPs but bear with me… we are getting somewhere)*

Monte-Carlo Policy Evaluation (3/5)

An **episode** sampled from an MDP under a policy π is a sequence of states, actions and rewards which ends in a terminal state:

$$s_1, a_1, r_1, s_2, a_2, r_2, s_3, a_3, r_3, \ldots, s_T$$

where s_i is the state at time i, a_i is the action taken at time i and r_i is the corresponding reward obtained at time i.

The return at time t for a concrete episode $s_1, a_1, r_1, s_2, a_2, r_2 \ldots, s_T$

$$g_t = r_1 + \gamma \cdot r_2 + \gamma^2 \cdot r_3 + \ldots = \sum_{i=0}^{T-1} r_i \cdot \gamma^i$$

We can have bounds ∞, just remember that all rewards after T are 0.

Monte-Carlo Policy Evaluation (3/5)

An **episode** sampled from an MDP under a policy π is a sequence of states, actions and rewards which ends in a terminal state:

$$s_1, a_1, r_1, s_2, a_2, r_2, s_3, a_3, r_3, \ldots, s_T$$

where s_i is the state at time i, a_i is the action taken at time i and r_i is the corresponding reward obtained at time i.

The return at time t for a concrete episode $s_1, a_1, r_1, s_2, a_2, r_2 \ldots, s_T$

$$g_t = r_1 + \gamma \cdot r_2 + \gamma^2 \cdot r_3 + \ldots = \sum_{i=0}^{T-1} r_i \cdot \gamma^i$$

Monte-Carlo Policy Evaluation (3/5)

An **episode** sampled from an MDP under a policy π is a sequence of states, actions and rewards which ends in a terminal state:

$$s_1, a_1, r_1, s_2, a_2, r_2, s_3, a_3, r_3, \ldots, s_T$$

where s_i is the state at time i, a_i is the action taken at time i and r_i is the corresponding reward obtained at time i.

The return at time t for a concrete episode $s_1, a_1, r_1, s_2, a_2, r_2 \ldots, s_T$

$$g_t = r_1 + \gamma \cdot r_2 + \gamma^2 \cdot r_3 + \ldots = \sum_{i=0}^{T-1} r_i \cdot \gamma^i$$

Monte-Carlo Policy Evaluation (3/5)

An **episode** sampled from an MDP under a policy π is a sequence of states, actions and rewards which ends in a terminal state:

$$s_1, a_1, r_1, s_2, a_2, r_2, s_3, a_3, r_3, \ldots, s_T$$

where s_i is the state at time i, a_i is the action taken at time i and r_i is the corresponding reward obtained at time i.

The return at time t for a concrete episode $s_1, a_1, r_1, s_2, a_2, r_2 \ldots, s_T$

$$g_t = r_1 + \gamma \cdot r_2 + \gamma^2 \cdot r_3 + \ldots = \sum_{i=0}^{T-1} r_i \cdot \gamma^i$$

Monte-Carlo Policy Evaluation (4/5)

We will now try to approximate $V^{\pi}(s)$ directly using $V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \mid X_t = s]$ using sampled episodes. *After all, expectation can be approximated by an average of sampled values.*

We will sample finite episodes (after all we can't sample infinitely long episodes in practice). *This also means that MC policy estimation can only be used for episodic RL problems.*

Monte-Carlo Policy Evaluation (5/5)

Why the problem is not straightforward: *If we only wanted to estimate* $\mathbb{E}[G_t]$, *that would be easy, but we want to estimate* $\mathbb{E}[G_t \mid X_t = s]$ *that is we need to condition… but we cannot condition arbitrarily… we can only observe episodes sampled under the given policy…* ***so we will need to "wait" for*** s ***to occur.***

We will see two different MC algorithms to do that: First-Visit MC Estimation and Every-Visit MC Estimation.

Monte-Carlo Policy Evaluation (5/5)

Why the problem is not straightforward: *If we only wanted to estimate* $\mathbb{E}[G_t]$, *that would be easy, but we want to estimate* $\mathbb{E}[G_t \mid X_t = s]$ *that is we need to condition… but we cannot condition arbitrarily… we can only observe episodes sampled under the given policy…* ***so we will need to "wait" for*** s ***to occur.***

We will see two different MC algorithms to do that: First-Visit MC Estimation and Every-Visit MC Estimation.

Monte-Carlo Policy Evaluation (5/5)

Why the problem is not straightforward: *If we only wanted to estimate* $\mathbb{E}[G_t]$, *that would be easy, but we want to estimate* $\mathbb{E}[G_t \mid X_t = s]$ *that is we need to condition… but we cannot condition arbitrarily… we can only observe episodes sampled under the given policy…* ***so we will need to "wait" for*** s ***to occur.***

We will see two different MC algorithms to do that: First-Visit MC Estimation and Every-Visit MC Estimation.

Monte-Carlo Policy Evaluation (5/5)

Why the problem is not straightforward: *If we only wanted to estimate* $\mathbb{E}[G_t]$, *that would be easy, but we want to estimate* $\mathbb{E}[G_t \mid X_t = s]$ *that is we need to condition… but we cannot condition arbitrarily… we can only observe episodes sampled under the given policy…* ***so we will need to "wait" for*** s ***to occur.***

We will see two different MC algorithms to do that: First-Visit MC Estimation and Every-Visit MC Estimation.

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0,\ N(s) = 0,\ V^{\pi}(s) = \mathit{undefined}$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

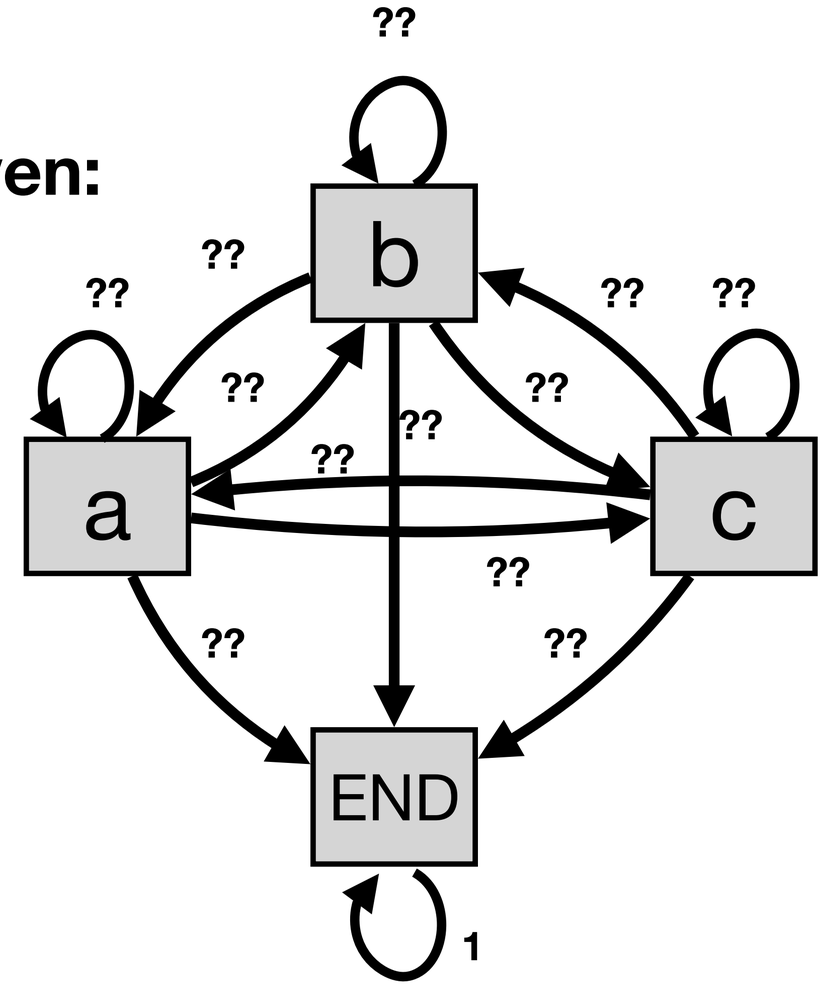
$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Recall Our Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{L, R\}$$

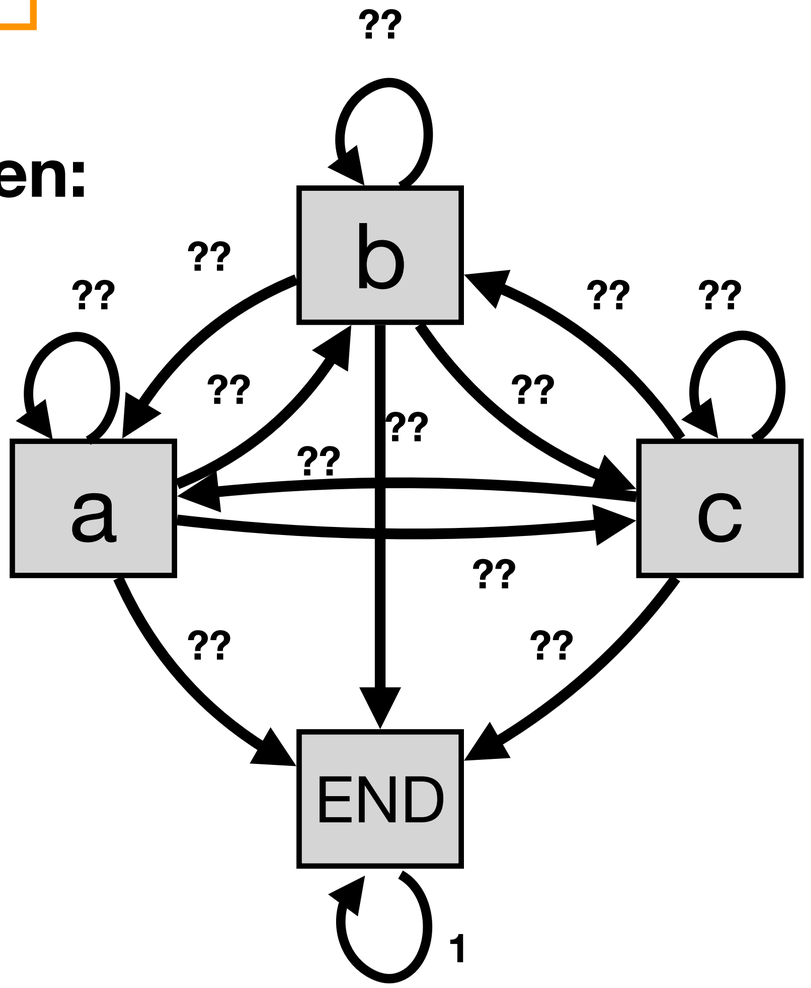
Some policy π is given (details not important now).

Recall Our Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{L, R\}$$

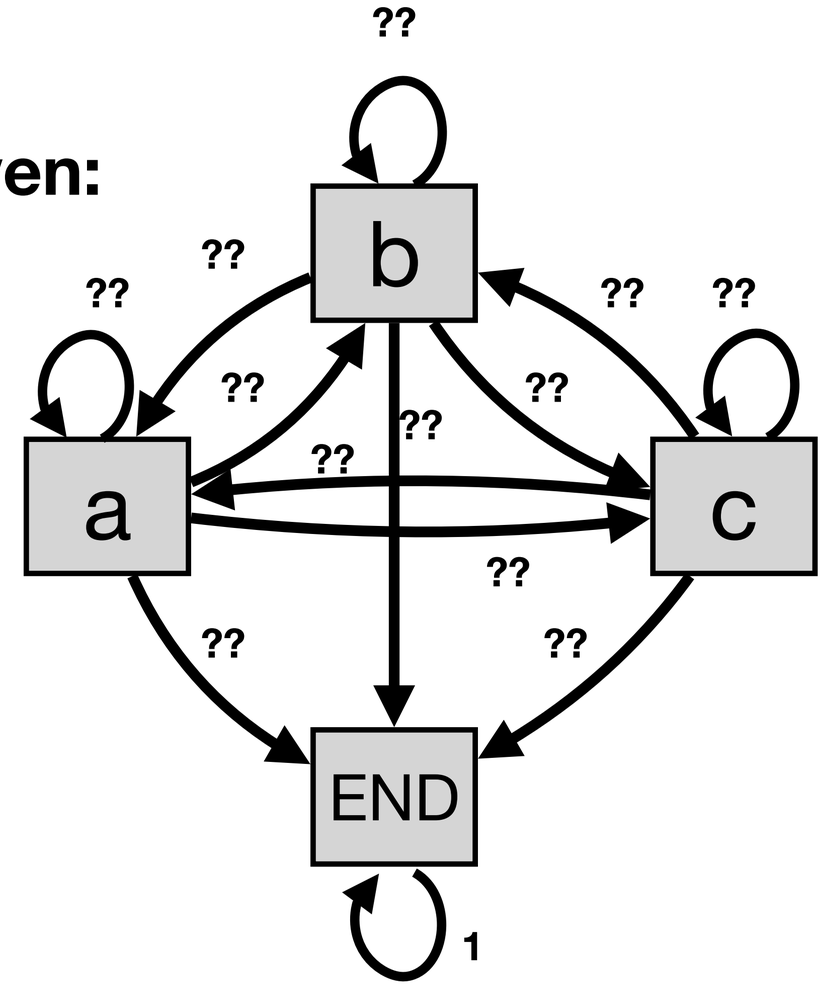
Some policy π is given (details not important now).

Recall Our Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{L, R\}$$

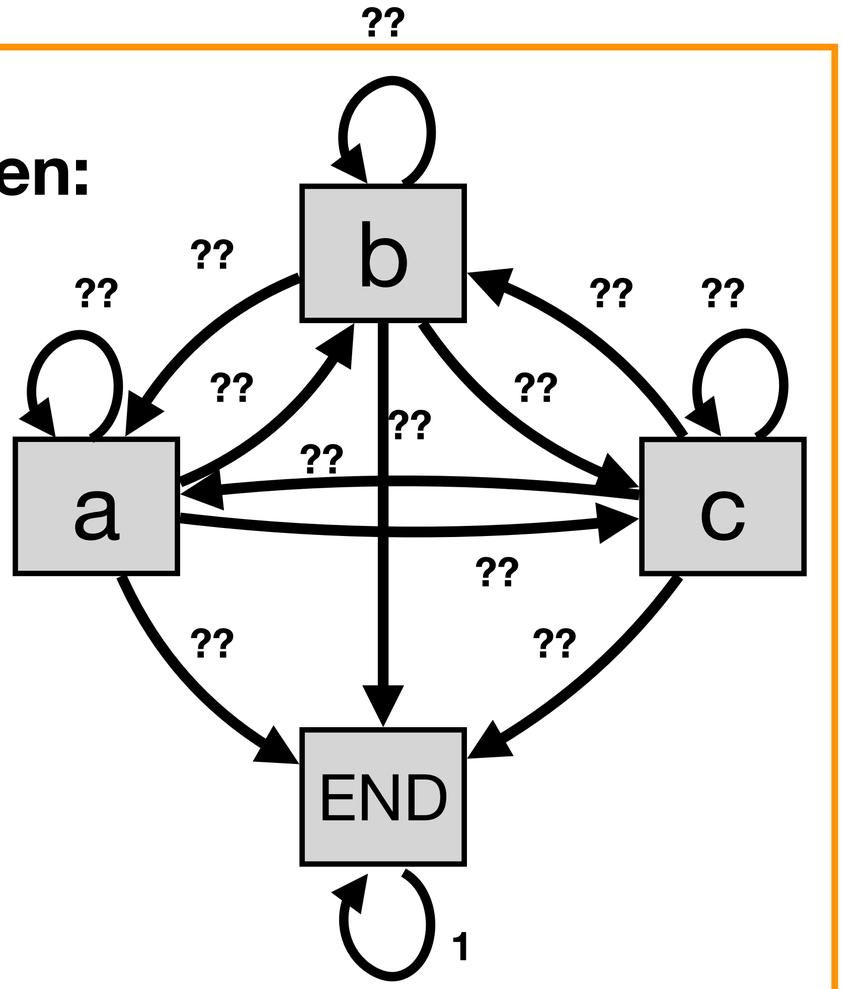
Some policy π is given (details not important now).

Recall Our Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{L, R\}$$

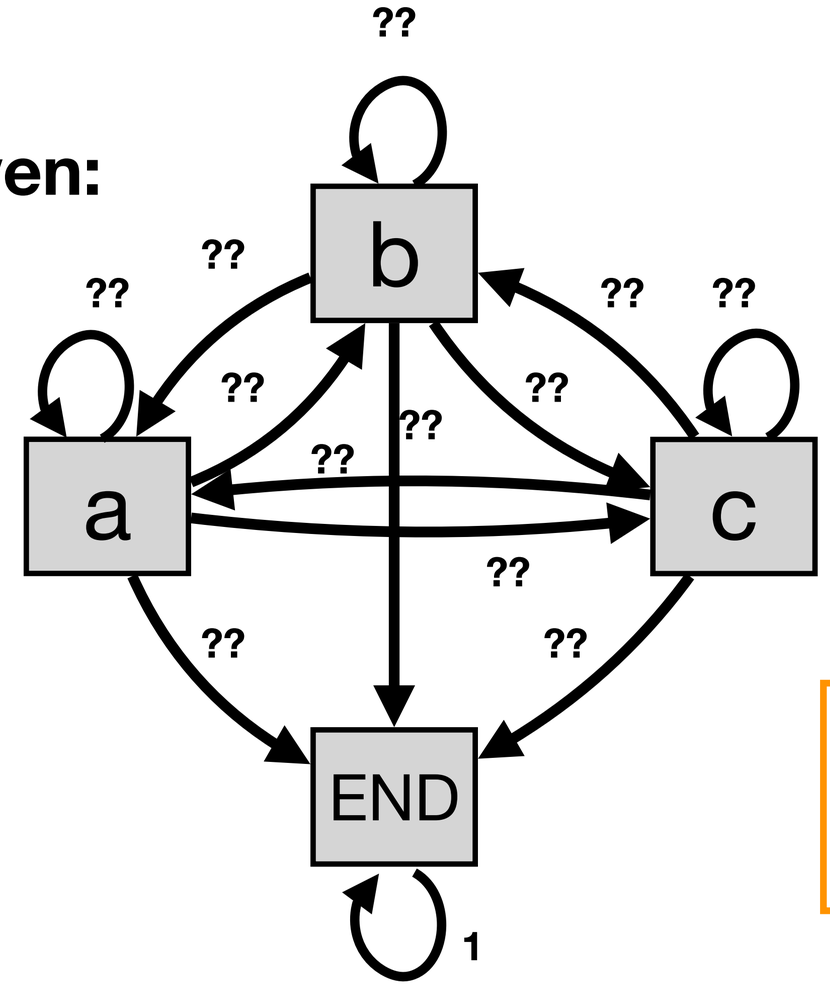
Some policy π is given (details not important now).

Recall Our Example

Agent: 

Rewards??

States are given:



Actions are given:

$$A = \{L, R\}$$

Some policy π is given (details not important now).

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^\pi(a) = 10, V^\pi(b) = 10, V^\pi(c) = 0, V^\pi(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^\pi(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^\pi(a) = 10, V^\pi(b) = 10, V^\pi(c) = 0, V^\pi(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^\pi(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^\pi(a) = 10, V^\pi(b) = 10, V^\pi(c) = 0, V^\pi(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^\pi(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 30, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 2, N(b) = 2, N(c) = 2, N(\textbf{end}) = 2$

$V^{\pi}(a) = 15, V^{\pi}(b) = 15, V^{\pi}(c) = 5, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 30, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 2, N(b) = 2, N(c) = 2, N(\textbf{end}) = 2$

$V^{\pi}(a) = 15, V^{\pi}(b) = 15, V^{\pi}(c) = 5, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 30, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 2, N(b) = 2, N(c) = 2, N(\textbf{end}) = 2$

$V^{\pi}(a) = 15, V^{\pi}(b) = 15, V^{\pi}(c) = 5, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 30, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 2, N(b) = 2, N(c) = 2, N(\textbf{end}) = 2$

$V^{\pi}(a) = 15, V^{\pi}(b) = 15, V^{\pi}(c) = 5, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 30, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 2, N(b) = 2, N(c) = 2, N(\textbf{end}) = 2$

$V^{\pi}(a) = 15, V^{\pi}(b) = 15, V^{\pi}(c) = 5, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

First-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 1, N(c) = 1, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 30, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 2, N(b) = 2, N(c) = 2, N(\textbf{end}) = 2$

$V^{\pi}(a) = 15, V^{\pi}(b) = 15, V^{\pi}(c) = 5, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

If t is the first occurrence of state s in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

~~**If** t is the first occurrence of state s in the episode e_i~~ /* This was for first-visit MC */

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,1}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit Monte-Carlo Evaluation

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

~~**If** t is the first occurrence of state s in the episode e_i~~ /* This was for first-visit MC */

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,1}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0,$ end

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^\pi(a) = 10, V^\pi(b) = 5, V^\pi(c) = 0, V^\pi(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 40, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 3, N(b) = 4, N(c) = 3, N(\textbf{end}) = 2$

$V^\pi(a) = 10, V^\pi(b) = 10, V^\pi(c) = \frac{10}{3}, V^\pi(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^\pi(s) := G(s)/N(s)$ /* *Update current estimate* */

Every-Visit MC Evaluation (Example)

Given: $S = \{a, b, c, \text{end}\}, A = \{L, R\}, \gamma = 1$

Sampled episodes (using given policy π):

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0, \text{end}$

$e_2 = a, R, 0, b, R, 10, c, L, 0, b, R, 10, a, L, 0, \text{end}$

After iteration 1:

$G(a) = 10, G(b) = 10, G(c) = 0, G(\textbf{end}) = 0$

$N(a) = 1, N(b) = 2, N(c) = 2, N(\textbf{end}) = 1$

$V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

After iteration 2:

$G(a) = 30, G(b) = 40, G(c) = 10, G(\textbf{end}) = 0$

$N(a) = 3, N(b) = 4, N(c) = 3, N(\textbf{end}) = 2$

$V^{\pi}(a) = 10, V^{\pi}(b) = 10, V^{\pi}(c) = \frac{10}{3}, V^{\pi}(\textbf{end}) = 0$

Initialize: $G(s) = 0, N(s) = 0$ **for all** $s \in S$.

For $i = 1, \ldots, N$:

Sample episode

$e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s) := G(s) + g_{i,t}$ /* *Increment total return counter* */

$V^{\pi}(s) := G(s)/N(s)$ /* *Update current estimate* */

Statistical Properties (1/7)

- First-visit MC Policy Evaluation is **unbiased** (and hence also consistent) estimator.
- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has better MSE.

Optional: Statistical Properties (2/7)

First-visit MC Policy Evaluation is **unbiased** (and hence also consistent) estimator.

Proof Sketch:

Assuming Markov property, the first occurrence* of the state s at time t together with the subsequence starting at t gives us an unbiased estimate of the return starting from s (this is practically from definition), i.e., $\mathbb{E}[G_t^\pi \mid X_t = s]$, which is by definition equal to $V^\pi(s)$. First-visit MC averages such independent samples from different episodes (different episodes => independence).

**Do you see why we cannot take, e.g., the last occurrence? Hint: Are subsequences starting with the last occurrence of s special in some way?*

Optional: Statistical Properties (2/7)

First-visit MC Policy Evaluation is **unbiased** (and hence also consistent) estimator.

Proof Sketch:

Assuming Markov property, the first occurrence* of the state s at time t together with the subsequence starting at t gives us an unbiased estimate of the return starting from s (this is practically from definition), i.e., $\mathbb{E}[G_t^\pi \mid X_t = s]$, which is by definition equal to $V^\pi(s)$. First-visit MC averages such independent samples from different episodes (different episodes => independence).

**Do you see why we cannot take, e.g., the last occurrence? Hint: Are subsequences starting with the last occurrence of s special in some way?*

Optional: Statistical Properties (2/7)

First-visit MC Policy Evaluation is **unbiased** (and hence also consistent) estimator.

Proof Sketch:

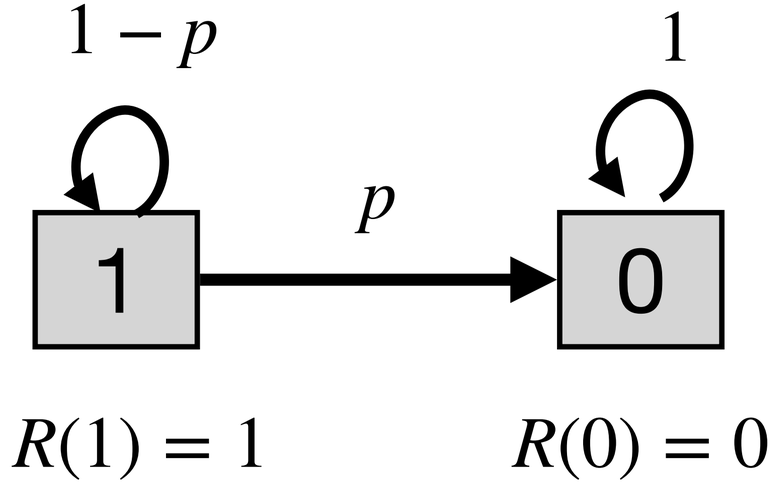
Assuming Markov property, the first occurrence* of the state s at time t together with the subsequence starting at t gives us an unbiased estimate of the return starting from s (this is practically from definition), i.e., $\mathbb{E}[G_t^\pi \mid X_t = s]$, which is by definition equal to $V^\pi(s)$. First-visit MC averages such independent samples from different episodes (different episodes => independence).

**Do you see why we cannot take, e.g., the last occurrence? Hint: Are subsequences starting with the last occurrence of s special in some way?*

Optional: Statistical Properties (3/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

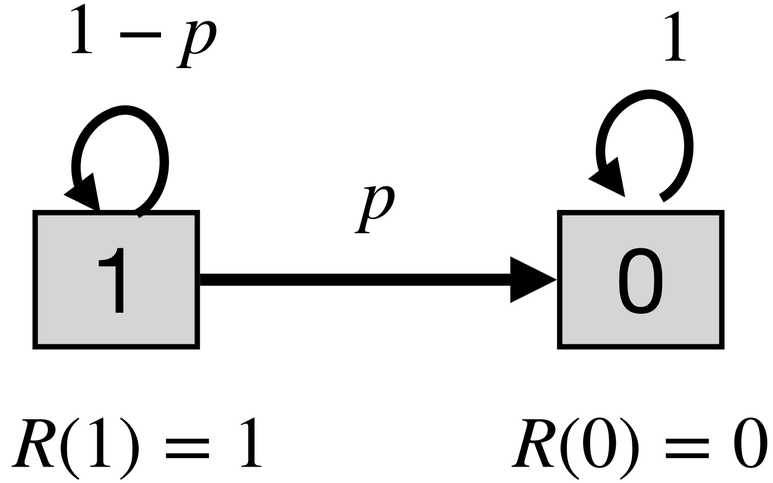


$\gamma = 1$

Optional: Statistical Properties (3/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

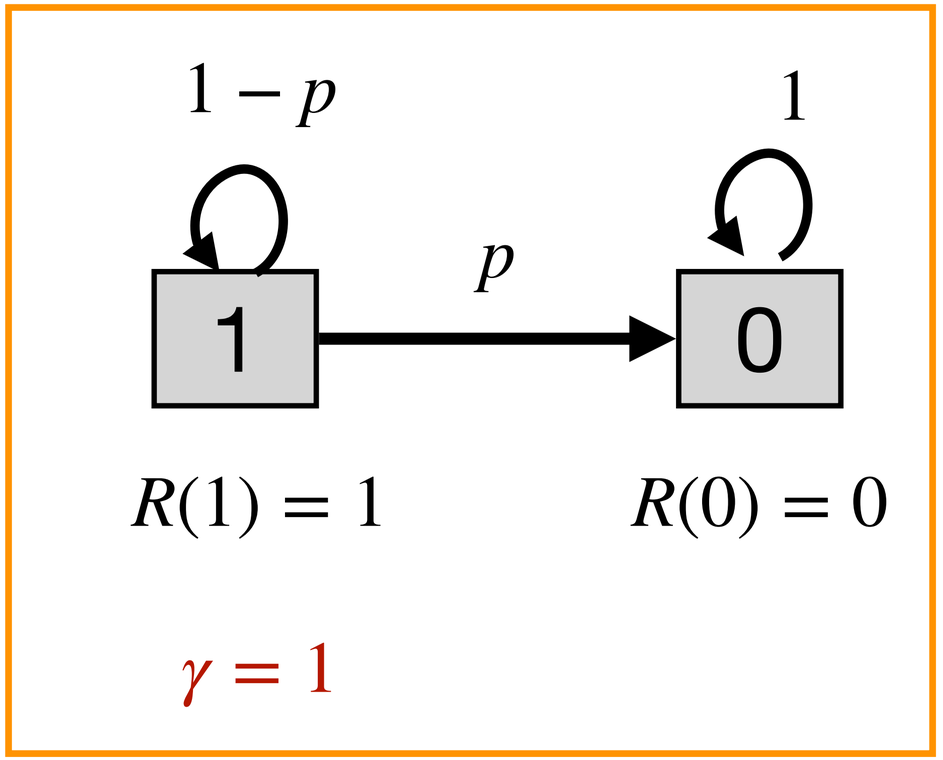


$\gamma = 1$

Optional: Statistical Properties (3/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

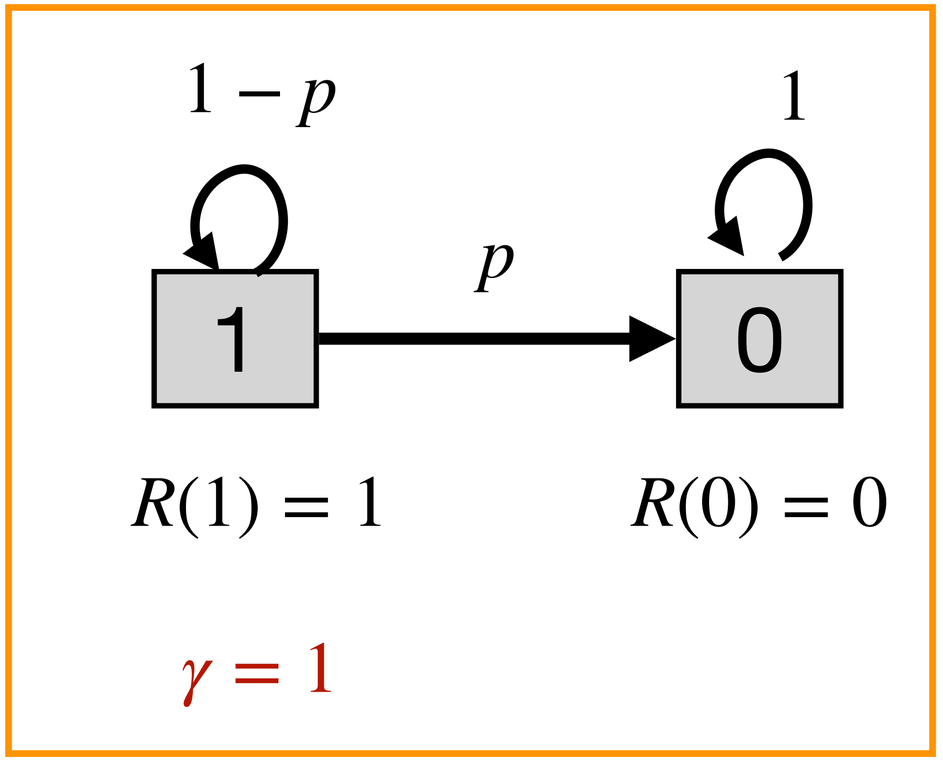
- **Example (Showing that it is biased):**



Optional: Statistical Properties (3/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

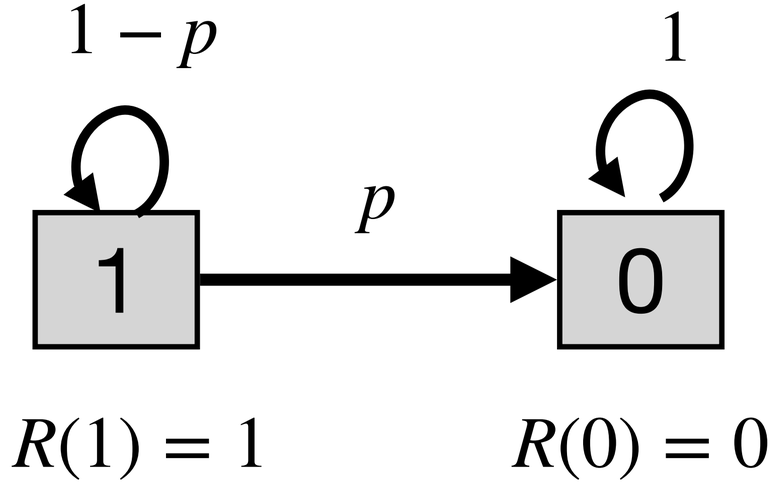


Optional: Statistical Properties (4/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

  - Computing V explicitly using Bellman equation:
    $V(1) = 1 + (1-p) \cdot V(1) + p \cdot 0$
    Hence, $V(1) = \frac{1}{p}$.



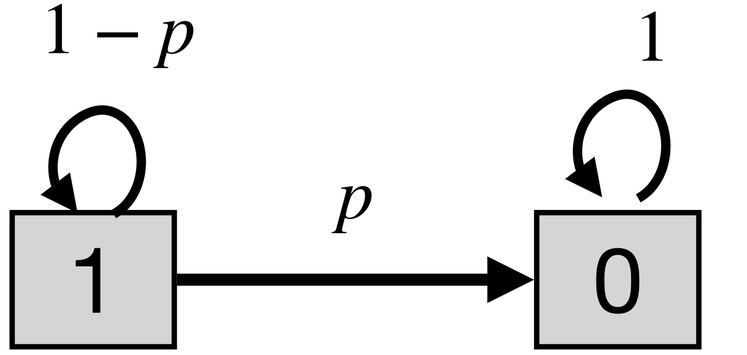
$\gamma = 1$

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

- First-Visit MC:

$$\mathbb{E}[\hat{V}_{FV}(1)] = p + 2(1-p)p + 3(1-p)^2 p + \ldots = p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n = p \cdot \frac{1}{p^2} = \frac{1}{p}$$

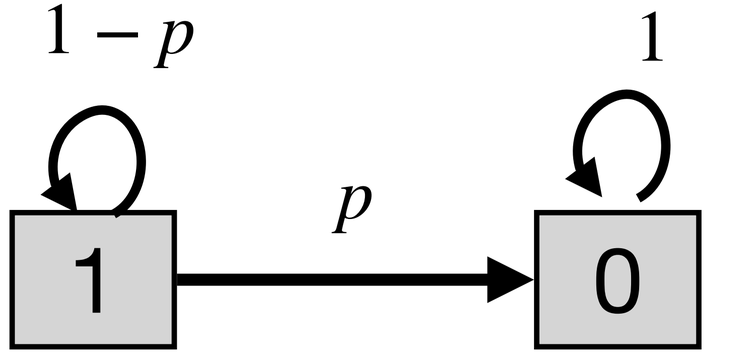
UNBIASED

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

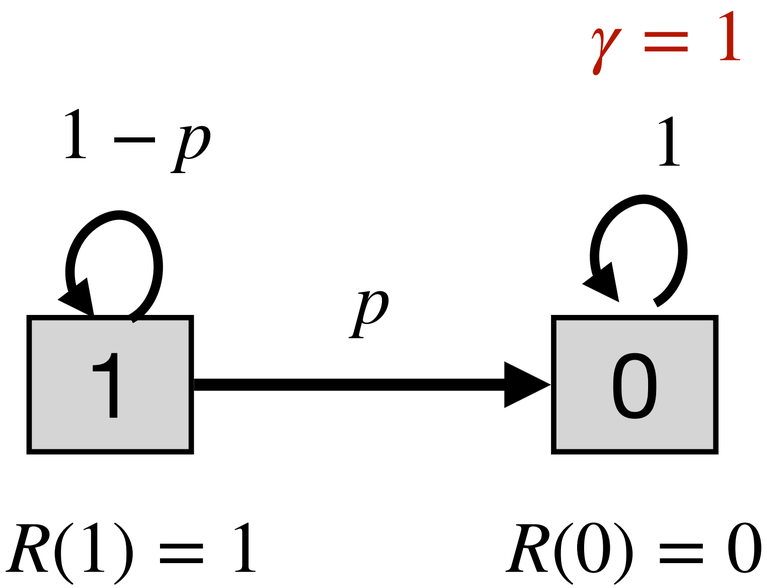
- First-Visit MC:

$$\boxed{\mathbb{E}[\hat{V}_{FV}(1)]} = p + 2(1-p)p + 3(1-p)^2 p + \ldots = p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n = p \cdot \frac{1}{p^2} = \frac{1}{p}$$

UNBIASED

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
  - Exact answer: $V(1) = \frac{1}{p}$.



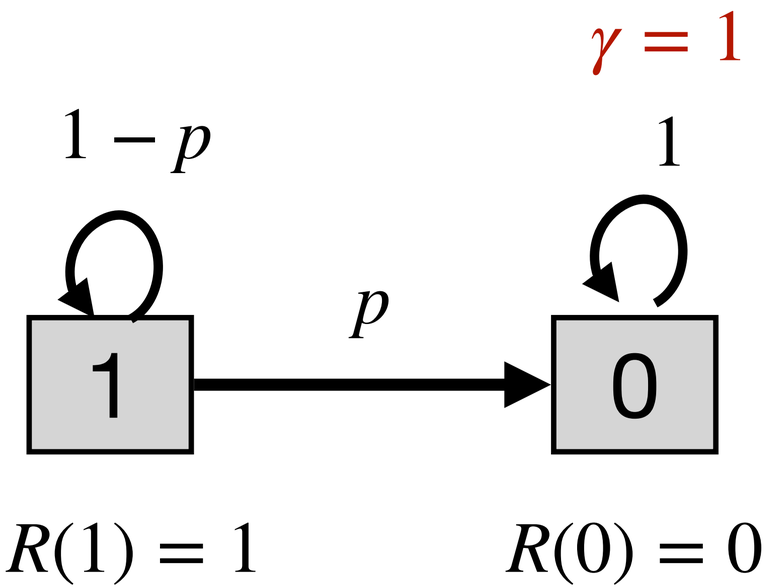
- First-Visit MC:

$$\mathbb{E}[\hat{V}_{FV}(1)] = p + 2(1-p)p + 3(1-p)^2 p + \ldots = p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n = p \cdot \frac{1}{p^2} = \frac{1}{p}$$

UNBIASED

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
  - Exact answer: $V(1) = \frac{1}{p}$.



- First-Visit MC:

$$\mathbb{E}[\hat{V}_{FV}(s)] = p + \boxed{2(1-p)p} + 3(1-p)^2 p + \ldots = p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n = p \cdot \frac{1}{p^2} = \frac{1}{p}$$

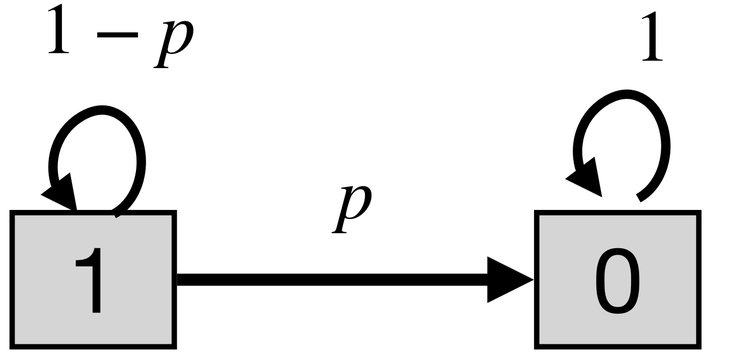
UNBIASED

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

- First-Visit MC:

$$\mathbb{E}[\hat{V}_{FV}(s)] = p + 2(1-p)p + \boxed{3(1-p)^2 p} + \ldots = p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n = p \cdot \frac{1}{p^2} = \frac{1}{p}$$

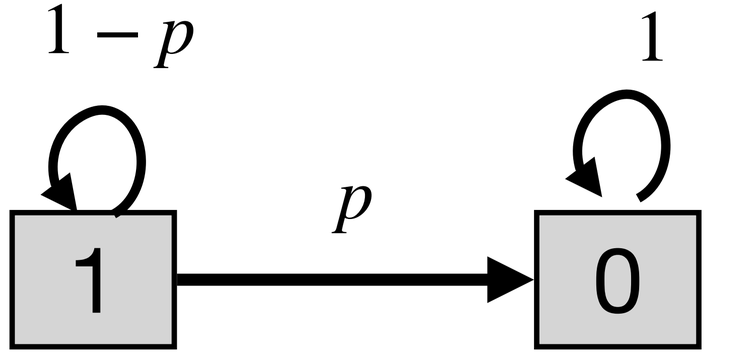
UNBIASED

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

- First-Visit MC:

$$\mathbb{E}[\hat{V}_{FV}(s)] = p + 2(1-p)p + 3(1-p)^2 p + \ldots = \boxed{p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n} = p \cdot \frac{1}{p^2} = \frac{1}{p}$$

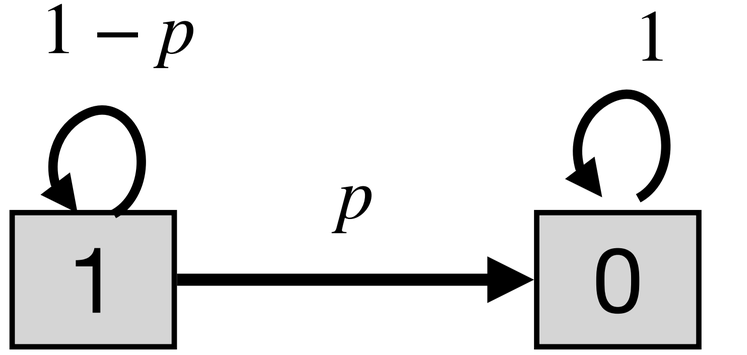
UNBIASED

Optional: Statistical Properties (5/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

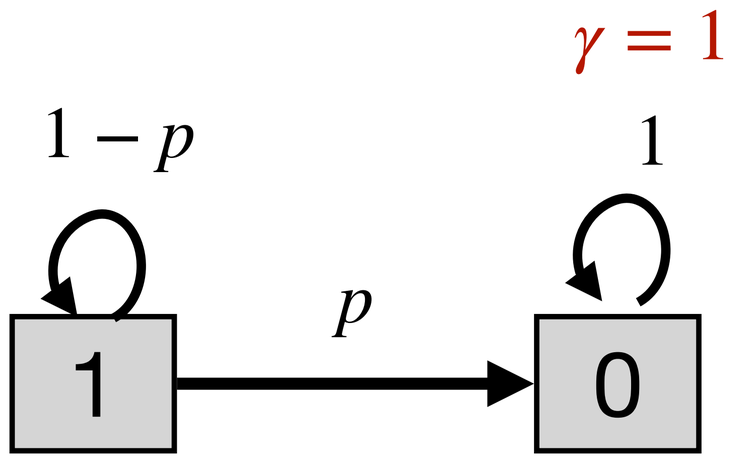
- First-Visit MC:

$$\mathbb{E}[\hat{V}_{FV}(s)] = p + 2(1-p)p + 3(1-p)^2 p + \ldots = p \sum_{n=0}^{\infty} (n+1) \cdot (1-p)^n = p \cdot \frac{1}{p^2} = \boxed{\frac{1}{p}}$$

UNBIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
  - Exact answer: $V(1) = \frac{1}{p}$.



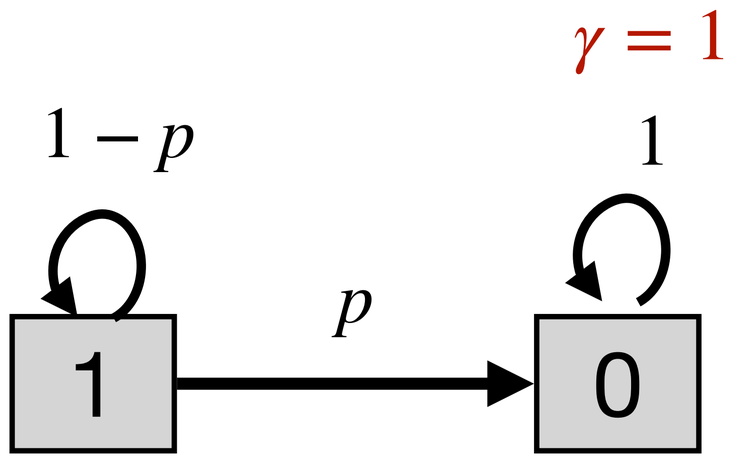
- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = p\sum_{n=0}^{\infty} \frac{n+2}{2} \cdot (1-p)^n = p \cdot \frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

BIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
  - Exact answer: $V(1) = \frac{1}{p}$.



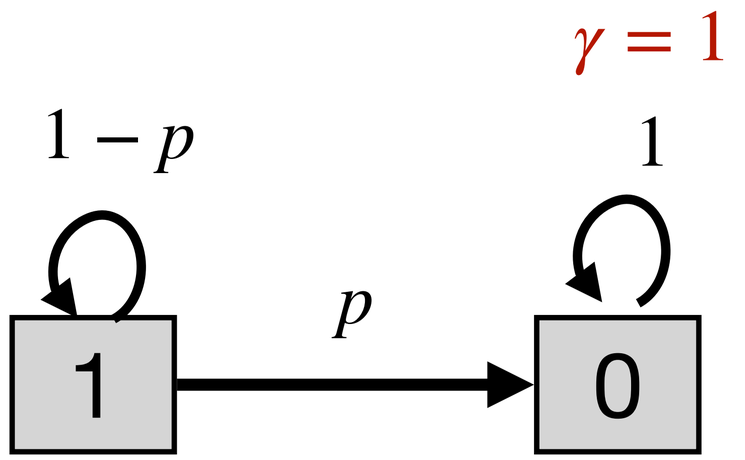
- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = p\sum_{n=0}^{\infty} \frac{n+2}{2} \cdot (1-p)^n = p \cdot \frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

BIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
- Exact answer: $V(1) = \frac{1}{p}$.



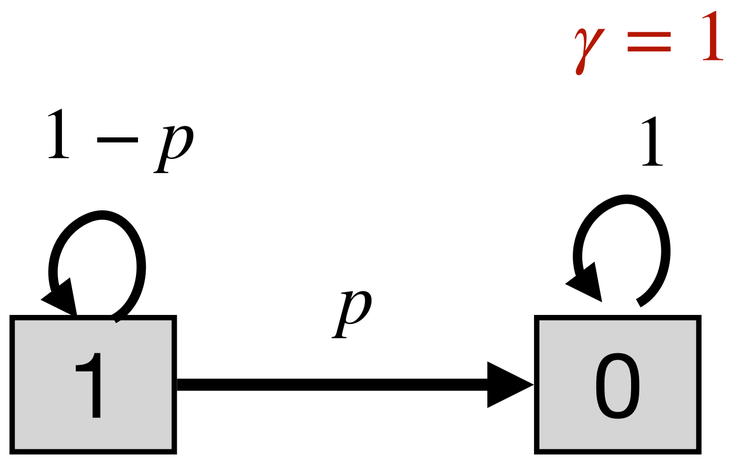
- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = p \sum_{n=0}^{\infty} \frac{n+2}{2} \cdot (1-p)^n = p \cdot \frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

BIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
  - Exact answer: $V(1) = \frac{1}{p}$.



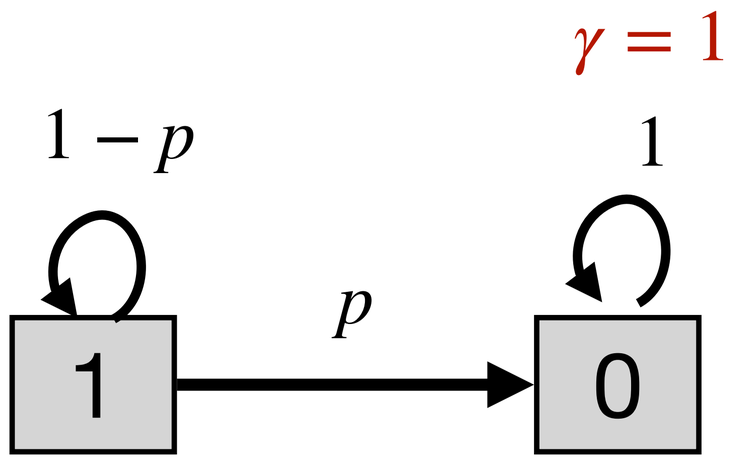
- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = p\sum_{n=0}^{\infty}\frac{n+2}{2}\cdot(1-p)^n = p\cdot\frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

BIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.
- **Example (Showing that it is biased):**
  - Exact answer: $V(1) = \frac{1}{p}$.



- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = p\sum_{n=0}^{\infty} \frac{n+2}{2} \cdot (1-p)^n = p \cdot \frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

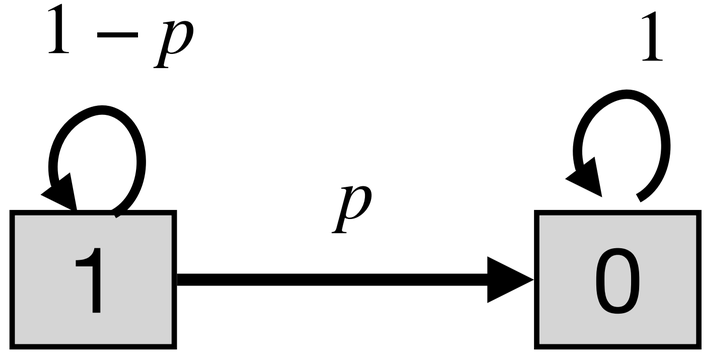
BIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = \boxed{p \sum_{n=0}^{\infty} \frac{n+2}{2} \cdot (1-p)^n} = p \cdot \frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

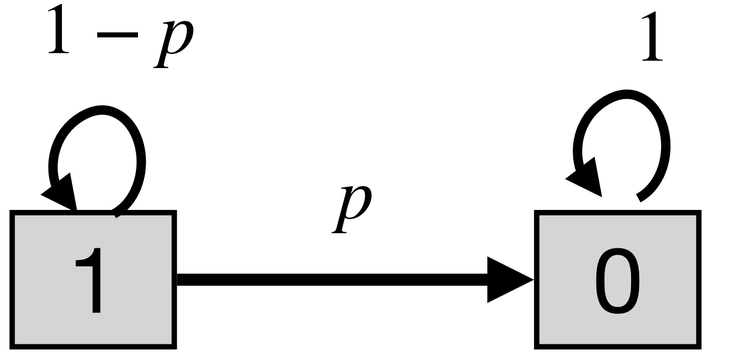
BIASED

Optional: Statistical Properties (6/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.

- **Example (Showing that it is biased):**

$\gamma = 1$



$R(1) = 1$ $R(0) = 0$

- Exact answer: $V(1) = \frac{1}{p}$.

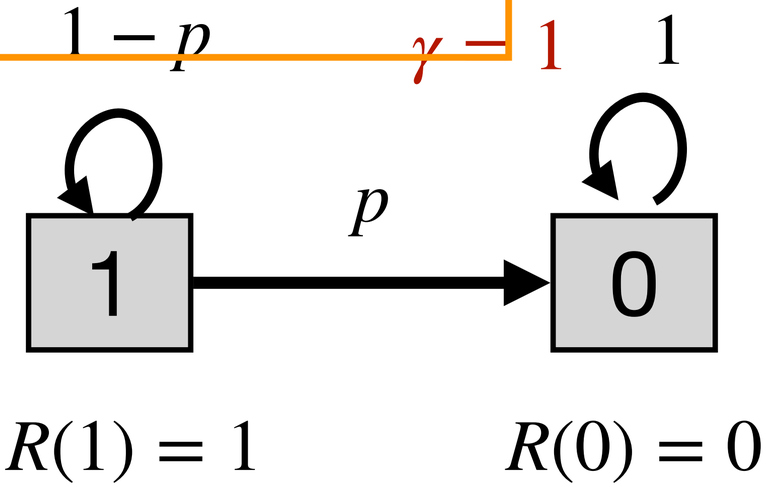
- Every-Visit MC (Bias):

$$\mathbb{E}[\hat{V}_{EV}(1)] = p + \frac{3}{2}(1-p)p + 2(1-p)^2 p + \ldots = p\sum_{n=0}^{\infty} \frac{n+2}{2} \cdot (1-p)^n = p \cdot \frac{p+1}{p^2} = \frac{p+1}{2p} \neq \frac{1}{p}$$

BIASED

Optional: Statistical Properties (7/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.



- **Example (Showing that it is biased):**
- Exact answer: $V(1) = \frac{1}{p}$.
- Every-Visit MC (**Consistency**):

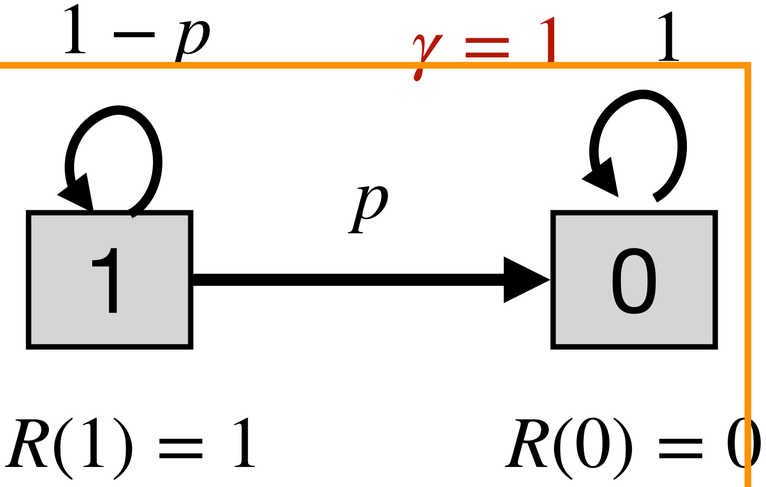
  $\hat{V}_{EV} = \frac{T+1}{2}$ where T is a geometrically distributed r.v. with expectation $\frac{1}{p}$.

  Averaging estimators over n independent episodes, one can show with a bit of algebraic manipulations that $P\left[\left|\hat{V}_n - \frac{1}{p}\right| < \varepsilon\right] = 1$ for all $0 < \varepsilon$.

Consistent

Optional: Statistical Properties (7/7)

- Every-visit MC Policy Evaluation is a **biased** but **consistent** estimator, which often has **better MSE**.



- **Example (Showing that it is biased):**
- Exact answer: $V(1) = \frac{1}{p}$.
- Every-Visit MC (**Consistency**):

  $\hat{V}_{EV} = \frac{T+1}{2}$ where T is a geometrically distributed r.v. with expectation $\frac{1}{p}$.

  Averaging estimators over n independent episodes, one can show with a bit of algebraic manipulations that $P\left[\left|\hat{V}_n - \frac{1}{p}\right| < \varepsilon\right] = 1$ for all $0 < \varepsilon$.

Consistent

Incremental Monte-Carlo Evaluation

Initialize: $N(s) = 0, V^{\pi}(s) = undefined$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$V^{\pi}(s) := V^{\pi}(S) + \alpha \cdot (g_{i,t} - V^{\pi}(s))$ /* *Update value function* */

Special case: When we use $\alpha = \frac{1}{N(s)}$ then the resulting incremental MC becomes equivalent to every-visit MC.

Incremental Monte-Carlo Evaluation

Initialize: $N(s) = 0, V^{\pi}(s) = undefined$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$V^{\pi}(s) := V^{\pi}(S) + \alpha \cdot (g_{i,t} - V^{\pi}(s))$ /* *Update value function* */

Special case: When we use $\alpha = \frac{1}{N(s)}$ then the resulting incremental MC becomes equivalent to every-visit MC.

Incremental Monte-Carlo Evaluation

Initialize: $N(s) = 0, V^{\pi}(s) = undefined$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$V^{\pi}(s) := V^{\pi}(S) + \alpha \cdot (g_{i,t} - V^{\pi}(s))$ /* *Update value function* */

Special case: When we use $\alpha = \frac{1}{N(s)}$ then the resulting incremental MC becomes equivalent to every-visit MC.

Incremental Monte-Carlo Evaluation

Initialize: $N(s) = 0, V^{\pi}(s) = undefined$ **for all** $s \in S$.

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$.

For each time step $1 \leq t \leq T_i$:

s is the state visited at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$V^{\pi}(s) := V^{\pi}(S) + \alpha \cdot (g_{i,t} - V^{\pi}(s))$ /* *Update value function* */

Special case: When we use $\alpha = \frac{1}{N(s)}$ then the resulting incremental MC becomes equivalent to every-visit MC.

Summary (So Far)

- **MC Methods:**
  - Try to estimate $V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \mid X_t = s]$ directly as an average over sampled episodes (which is also why they need the episodic settings).
  - They do not use the Markov assumption!
  - Converge to the true values.
  - Can have high variance and some of them are also biased (first-visit MC is one which is not biased).

Summary (So Far)

- **MC Methods:**
  - Try to estimate $V^{\pi}(s) = \mathbb{E}[G_t^{\pi} \mid X_t = s]$ directly as an average over sampled episodes (which is also why they need the episodic settings).
  - They do not use the Markov assumption!
  - Converge to the true values.
  - Can have high variance and some of them are also biased (first-visit MC is one which is not biased).

Part 4: Temporal Difference Learning

(We are still dealing with policy evaluation)

Temporal Difference Learning: A Teaser

- **TD learning** combines Monte-Carlo estimation and dynamic programming ideas.
- **TD learning** can be used both in episodic and infinite-horizon non-episodic settings,
- **TD learning** updates estimates of V^{π} continually, after every consecutive tuple *state-action-reward-state* (therefore we do not need to wait till the end of an episode).

....

Temporal Difference Learning: A Teaser

- **TD learning** combines Monte-Carlo estimation and dynamic programming ideas.

- **TD learning** can be used both in episodic and infinite-horizon non-episodic settings,

- **TD learning** updates estimates of V^{π} continually, after every consecutive tuple *state-action-reward-state* (therefore we do not need to wait till the end of an episode).

....

Temporal Difference Learning: A Teaser

- **TD learning** combines Monte-Carlo estimation and dynamic programming ideas.

- **TD learning** can be used both in episodic and infinite-horizon non-episodic settings,

- **TD learning** updates estimates of V^{π} continually, after every consecutive tuple *state-action-reward-state* (therefore we do not need to wait till the end of an episode).

....

Temporal Difference Learning: A Teaser

- **TD learning** combines Monte-Carlo estimation and dynamic programming ideas.

- **TD learning** can be used both in episodic and infinite-horizon non-episodic settings,

- **TD learning** updates estimates of V^{π} continually, after every consecutive tuple *state-action-reward-state* (therefore we do not need to wait till the end of an episode).

....

TD-Learning: Basic Idea

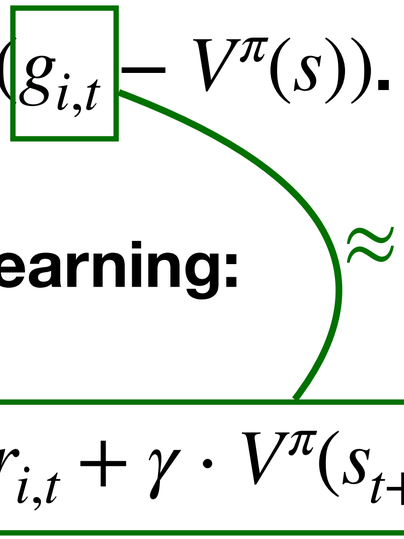
Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Incremental MC:

$$V^{\pi}(s) := V^{\pi}(s) + \alpha \cdot (g_{i,t} - V^{\pi}(s)).$$

Temporal Difference Learning:

$$V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t))$$



TD-Learning: Basic Idea

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Incremental MC:

$$V^\pi(s) := V^\pi(s) + \alpha \cdot (g_{i,t} - V^\pi(s)).$$

Temporal Difference Learning:

$\approx$

$$V^\pi(s_t) := V^\pi(s_t) + \alpha(r_{i,t} + \gamma \cdot V^\pi(s_{t+1}) - V^\pi(s_t))$$

TD-Learning: Basic Idea

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Incremental MC:

$$V^{\pi}(s) := V^{\pi}(s) + \alpha \cdot (g_{i,t} - V^{\pi}(s)).$$

Temporal Difference Learning:

$$V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t))$$

$\approx$

TD-Learning: Basic Idea

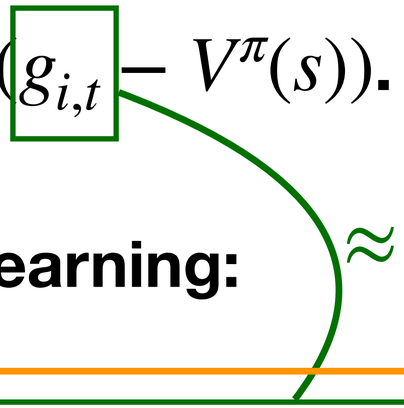
Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Incremental MC:

$$V^{\pi}(s) := V^{\pi}(s) + \alpha \cdot (g_{i,t} - V^{\pi}(s)).$$

Temporal Difference Learning:

$$V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t))$$



TD-Learning: Basic Idea

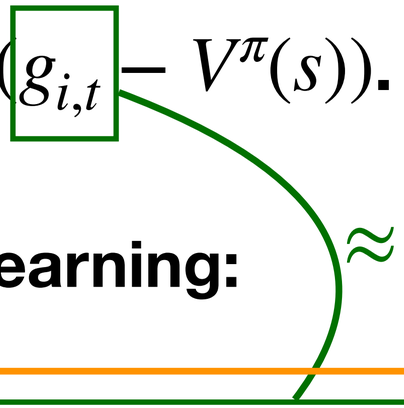
Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Incremental MC:

$$V^{\pi}(s) := V^{\pi}(s) + \alpha \cdot (g_{i,t} - V^{\pi}(s)).$$

Temporal Difference Learning:

$$V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t))$$



TD-Learning: Basic Idea

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Incremental MC:

$$V^\pi(s) := V^\pi(s) + \alpha \cdot (g_{i,t} - V^\pi(s)).$$

Temporal Difference Learning:

$$V^\pi(s_t) := V^\pi(s_t) + \alpha (r_{i,t} + \gamma \cdot V^\pi(s_{t+1}) - V^\pi(s_t))$$

TD-Learning: Relationship to Bellman Backup

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Bellman equation update rule:

$$V^{\pi}_{k+1}(s) := R(s, \pi(s)) + \gamma \cdot \sum_{s' \in S} P(s' \,|\, s, \pi(s)) \cdot V^{\pi}_{k}(s')$$

Temporal Difference Learning update rule:

$$\begin{aligned} V^{\pi}(s_t) &:= V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t)) \\ &= (1 - \alpha) \cdot V^{\pi}(s_t) + \alpha \cdot (r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})) \end{aligned}$$

TD-Learning: Relationship to Bellman Backup

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Bellman equation update rule:

$$V^{\pi}_{k+1}(s) := R(s, \pi(s)) + \gamma \cdot \sum_{s' \in S} P(s' \,|\, s, \pi(s)) \cdot V^{\pi}_{k}(s')$$

Temporal Difference Learning update rule:

$$\begin{aligned} V^{\pi}(s_t) &:= V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t)) \\ &= (1 - \alpha) \cdot V^{\pi}(s_t) + \alpha \cdot (r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})) \end{aligned}$$

TD-Learning: Relationship to Bellman Backup

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Bellman equation update rule:

$$V^{\pi}_{k+1}(s) := R(s, \pi(s)) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, \pi(s)) \cdot V^{\pi}_{k}(s')$$

Expectation

Temporal Difference Learning update rule:

$$\begin{aligned} V^{\pi}(s_t) &:= V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t)) \\ &= (1 - \alpha) \cdot V^{\pi}(s_t) + \alpha \cdot (r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})) \end{aligned}$$

Sample

TD-Learning: Relationship to Bellman Backup

Recall: $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

Bellman equation update rule:

$$V^{\pi}_{k+1}(s) := R(s, \pi(s)) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, \pi(s)) \cdot V^{\pi}_{k}(s')$$

Expectation

Temporal Difference Learning update rule:

$$\begin{aligned} V^{\pi}(s_t) &:= V^{\pi}(s_t) + \alpha(r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1}) - V^{\pi}(s_t)) \\ &= (1 - \alpha) \cdot V^{\pi}(s_t) + \alpha \cdot (r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})) \end{aligned}$$

Sample

TD-Learning: Pseudocode

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

TD-Learning: Pseudocode

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

TD-Learning: Pseudocode

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

TD-Learning: Pseudocode

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

TD-Learning: Example

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5,\ \gamma = 1$$

$$e_1 = a,\ R,\ 0,\ b,\ R,\ 10,\ c, L,\ 0,\ b,\ R,\ 0,\ c,\ R,\ 0,\ \text{end}$$

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^\pi(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^\pi(s_t) := V^\pi(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^\pi(s_{t+1})}_{\text{TD target}} - V^\pi(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^\pi(a) := 0,$

Iteration 2: $V^\pi(b) := 5,$

Iteration 3: $V^\pi(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^\pi(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^\pi(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$$\alpha = 0.5, \gamma = 1$$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

TD-Learning: Example 

Initialize: $V^{\pi}(s) = 0$ for all $s \in S$

Loop:

Sample tuple (s_t, a_t, r_t, s_{t+1}).

Update $V^{\pi}(s_t) := V^{\pi}(s_t) + \alpha \cdot (\underbrace{r_{i,t} + \gamma \cdot V^{\pi}(s_{t+1})}_{\text{TD target}} - V^{\pi}(s_t))$

$\alpha = 0.5, \gamma = 1$

$e_1 = a, R, 0, b, R, 10, c, L, 0, b, R, 0, c, R, 0,$ end

Iteration 1: $V^{\pi}(a) := 0,$

Iteration 2: $V^{\pi}(b) := 5,$

Iteration 3: $V^{\pi}(c) := 0.5(0 + 5) = 2.5,$

Iteration 4: $V^{\pi}(b) := 5 + 0.5 \cdot (0 + 2.5 - 5) = 3.75,$

Iteration 5: $V^{\pi}(c) := 2.5 + 0.5 \cdot (0 + 0 - 2.5) = 1.25.$

Every-Visit Monte-Carlo: $V^{\pi}(a) = 10, V^{\pi}(b) = 5, V^{\pi}(c) = 0, V^{\pi}(\textbf{end}) = 0$

What About the α's?

- One thing we can do is to have α depend on the number of iterations so far, i.e., we can have α_k instead of just α.

- Convergence is guaranteed when $\alpha_k's$ satisfy the following conditions (follows from Robbins-Munro algorithm):

$$\sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} a_k^2 < \infty.$$

- A sequence which satisfies the above conditions is, e.g., $a_k = \frac{1}{k}$. However, in practice, similar sequences do not have to converge very fast…

- *Note: It was also proved by Sutton (1988) that, for tabular MDPs, there always exists some small enough learning rate α such that TD converges but this result is not very practical.*

What About the α's?

- One thing we can do is to have α depend on the number of iterations so far, i.e., we can have α_k instead of just α.

- Convergence is guaranteed when $\alpha_k's$ satisfy the following conditions (follows from Robbins-Munro algorithm):

$$\sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} a_k^2 < \infty.$$

- A sequence which satisfies the above conditions is, e.g., $a_k = \frac{1}{k}$. However, in practice, similar sequences do not have to converge very fast…

- *Note: It was also proved by Sutton (1988) that, for tabular MDPs, there always exists some small enough learning rate α such that TD converges but this result is not very practical.*

What About the α's?

- One thing we can do is to have α depend on the number of iterations so far, i.e., we can have α_k instead of just α.

- Convergence is guaranteed when $\alpha_k's$ satisfy the following conditions (follows from Robbins-Munro algorithm):

$$\sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} a_k^2 < \infty.$$

- A sequence which satisfies the above conditions is, e.g., $a_k = \frac{1}{k}$. However, in practice, similar sequences do not have to converge very fast…

- *Note: It was also proved by Sutton (1988) that, for tabular MDPs, there always exists some small enough learning rate α such that TD converges but this result is not very practical.*

What About the α's?

- One thing we can do is to have α depend on the number of iterations so far, i.e., we can have α_k instead of just α.

- Convergence is guaranteed when $\alpha_k's$ satisfy the following conditions (follows from Robbins-Munro algorithm):

$$\sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} a_k^2 < \infty.$$

- A sequence which satisfies the above conditions is, e.g., $a_k = \frac{1}{k}$. However, in practice, similar sequences do not have to converge very fast…

- *Note: It was also proved by Sutton (1988) that, for tabular MDPs, there always exists some small enough learning rate α such that TD converges but this result is not very practical.*

What About the α's?

- One thing we can do is to have α depend on the number of iterations so far, i.e., we can have α_k instead of just α.

- Convergence is guaranteed when $\alpha_k's$ satisfy the following conditions (follows from Robbins-Munro algorithm):

$$\sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} a_k^2 < \infty.$$

- A sequence which satisfies the above conditions is, e.g., $a_k = \frac{1}{k}$. However, in practice, similar sequences do not have to converge very fast…

- *Note: It was also proved by Sutton (1988) that, for tabular MDPs, there always exists some small enough learning rate α such that TD converges but this result is not very practical.*

Policy Evaluation: Summary

	DPCE	MC	TD
Can use w/out access to true MDP models	X	X	X
Usable in continuing (non-episodic) setting	X		X
Assumes Markov process	X		X
Converges to true value in limit[3]	X	X	X
Unbiased estimate of value		X	

- DPCE = Dynamic Programming w/certainty equivalence estimates, MC = Monte Carlo, TD = Temporal Difference

Table from slides by Prof. Emma Brunskill

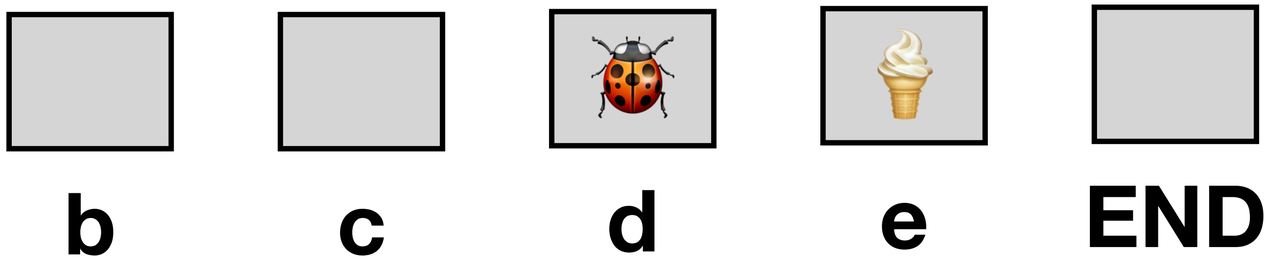
Part 5: Model-Free Control (Problem Statement)

Model-Free Control

- Given an MDP with unknown parameters (or generally an environment with which we can interact), **find an optimal policy π.**

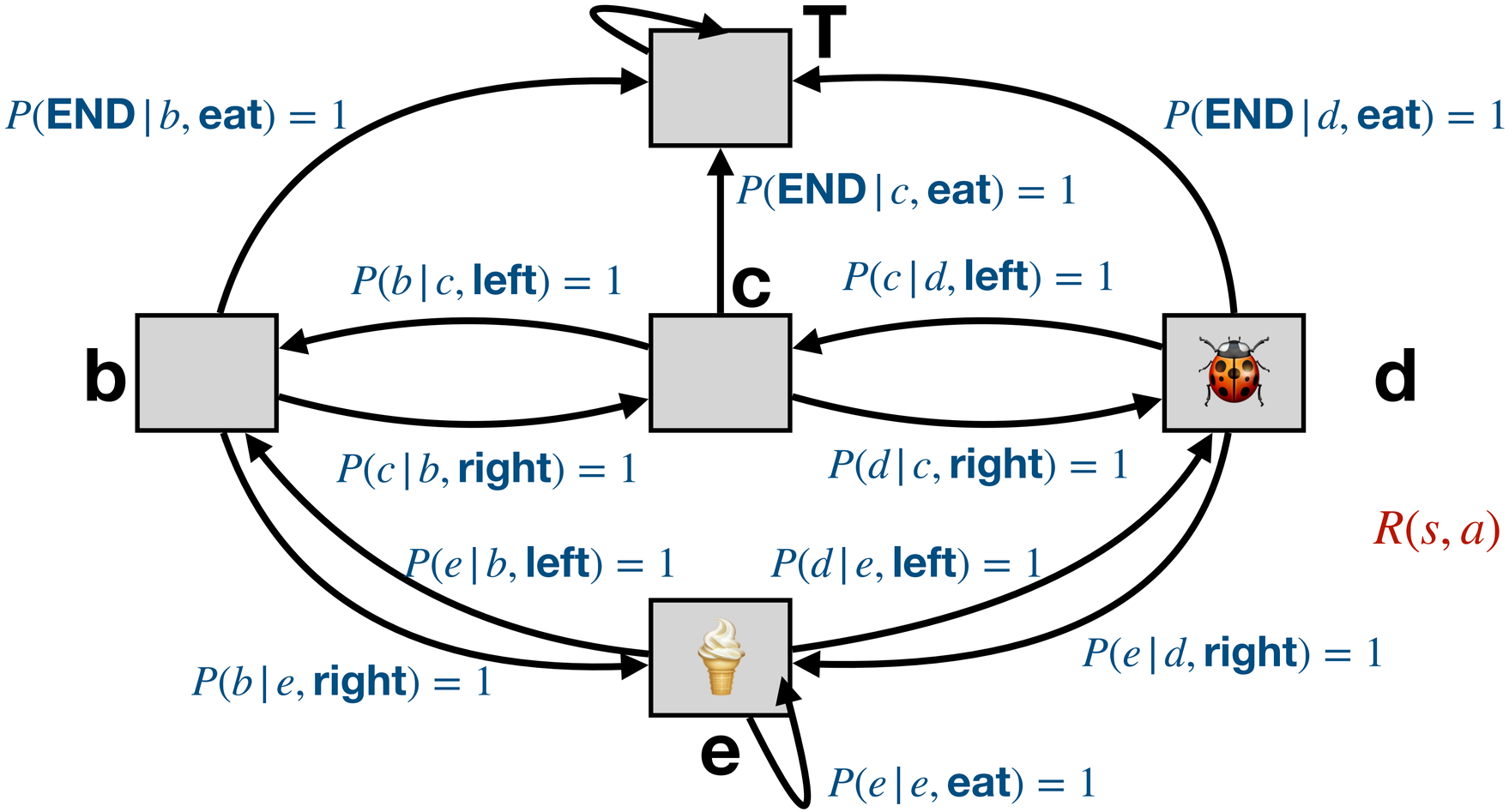
Running Example

- **Example we will use:**
  - Agent (ladybug)
  - State space: $S = \{b, c, d, e, \text{END}\}$, END is the terminal state.
  - Action space: $A = \{\text{left}, \text{right}, \text{eat}\}$.
  - We **do not know** $P(s' \mid s, a)$, $R(s, a)$ and $\pi(a \mid s)$.
  - We want to learn some optimal policy!



Running Example

- Here, is what the system will behave like - this is just for you to have some intuition, the RL algorithm will not have access to this information.



$$R(s,a) = \begin{cases} 1 & s \in \{b, c, d\} \\ 10 & s = e \textbf{ and } a = \textbf{eat} \\ 0 & \textbf{otherwise} \end{cases}$$

Part 6: Model-Free Policy Iteration

An Idea

- What if we wanted to use policy iteration to find the optimal policy?
- What would we need?
- **Answer:** We would need to be able to compute the state-action value function $Q^{\pi}(s, a)$ for any policy π. But that's not possible because we do not know the parameters of the MDP…
- **Idea:** Could we estimate $Q^{\pi}(s, a)$ in a similar way as we were estimating $V^{\pi}(s)$ last week? And then use policy improvement on that estimated $Q^{\pi}(s, a)$?

MC Estimation of $Q^{\pi}(s, a)$

Last time we talked about MC Estimation of the value function.

We can use the same idea for the estimation of the state-action value function $Q^{\pi}(s, a)$…

…then use that estimated $Q^{\pi}(s, a)$ as in policy iteration…

MC Estimation of $Q^\pi(s, a)$

Last time we talked about MC Estimation of the value function.

We can use the same idea for the estimation of the state-action value function $Q^\pi(s, a)$…

…then use that estimated $Q^\pi(s, a)$ as in policy iteration…

…and see how it fails if done naively.

A Naive Idea

- **THIS WILL NOT WORK (YET):**

Initialize: $G(s,a) = 0$, $N(s,a) = 0$ **for all** $s \in S$, $\pi_1 = \pi$ (*the given policy*)**.**

For $i = 1,\ldots,N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$ using π_i.

For each time step $1 \leq t \leq T_i$:

(**If** t is the first occurrence of state s followed by the action a in the episode e_i - Use this if you want first-visit MC)

s_t is the state visited at time t in the episode e_i

a_t is the action taken at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s_t, a_t) := G(s_t, a_t) + g_{i,1}$ /* *Increment total return counter* */

$Q(s_t, a_t) := G(s_t, a_t)/N(s_t, a_t)$ /* *Update current estimate* */

Set π_{i+1} *= greedy policy w.r.t.* Q*, i.e.,* $\pi(s) = \arg\max_{a \in A} Q(s,a)$ /* *breaking ties consistently* */.

Let's see why it will not work!

$S = \{b, c, d, e, \text{END}\}, A = \{\text{left}, \text{right}, \text{eat}\}$

$\pi(b) = \pi(c) = \pi(e) = \text{left}, \pi(d) = \text{eat}$

$e_1 = c, \text{left}, 1, b, \text{left}, 1, e, \text{left}, 1, d, \text{eat}, 0, \text{END}$

How can we ever estimate, e.g., $Q^{\pi}(b, \text{right})$??

The problem is we may never update the estimate for $Q^{\pi}(b, right)$ because the action taken in the state b is always left.

- **A simple idea** (that will not work yet… and will illustrate why we need to think about exploration)**:**
  - **THIS WILL NOT WORK (YET):**

    Initialize: $G(s, a) = 0, N(s, a) = 0$ **for all** $s \in S$.

    For $i = 1, \ldots, N$**:**

    Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$ using π.

    For each time step $1 \leq t \leq T_i$:

    (**If** t is the first occurrence of state s followed by the action a in the episode e_i - Use this if you want first-visit MC)

    s_t is the state visited at time t in the episode e_i

    a_t is the action taken at time t in the episode e_i

    $g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

    $N(s) := N(s) + 1$ /* *Increment total visits counter* */

    $G(s_t, a_t) := G(s_t, a_t) + g_{i,1}$ /* *Increment total return counter* */

    $Q^{\pi}(s_t, a_t) := G(s_t, a_t)/N(s_t, a_t)$ /* *Update current estimate* */

ε-Greedy Policy

- Given a Q-function $Q(s, a)$, we define the ε-greedy policy w.r.t. Q as

We assume ties are decided consistently

$$\pi(a \mid s) = \begin{cases} 1 - \varepsilon + \frac{\varepsilon}{|A|} & \text{when } a = \arg\max_{a \in A} Q(s, a) \\ \frac{\varepsilon}{|A|} & \text{when } a \neq \arg\max_{a \in A} Q(s, a) \end{cases}$$

MC On Policy Iteration

Initialize: $G(s,a) = 0,\ N(s,a) = 0,\ Q(s,a) = 0$ **for all** $s \in S, a \in A$**.**

Initialize: $\varepsilon = 1,\ k = 1$

For $i = 1, \ldots, N$**:**

Sample episode $e_i := s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$ **given** π_k**.**

For each time step $1 \leq t \leq T_i$:

(**If** t is the first occurrence of state s followed by the action a in the episode e_i - Use this if you want first-visit MC)

s_t is the state visited at time t in the episode e_i

a_t is the action taken at time t in the episode e_i

$g_{i,t} := r_{i,t} + \gamma \cdot r_{i,t+1} + \gamma^2 \cdot r_{i,t+2} + \ldots + \gamma^{T_i - t} \cdot r_{i,T_i}$

$N(s) := N(s) + 1$ /* *Increment total visits counter* */

$G(s_t, a_t) := G(s_t, a_t) + g_{i,1}$ /* *Increment total return counter* */

$Q(s_t, a_t) := G(s_t, a_t)/N(s_t, a_t)$ /* *Update current estimate* *

EndFor

$k = k + 1,\ \varepsilon = 1/k$

$\pi_k = \varepsilon-$**greedy policy w.r.t.** Q

Running Example (Initialization)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$

b **c** **d** **e** **END**

G(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

N(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$

$e_1 = d$

$$\pi_1(a \mid d) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$

$e_1 = d$, right

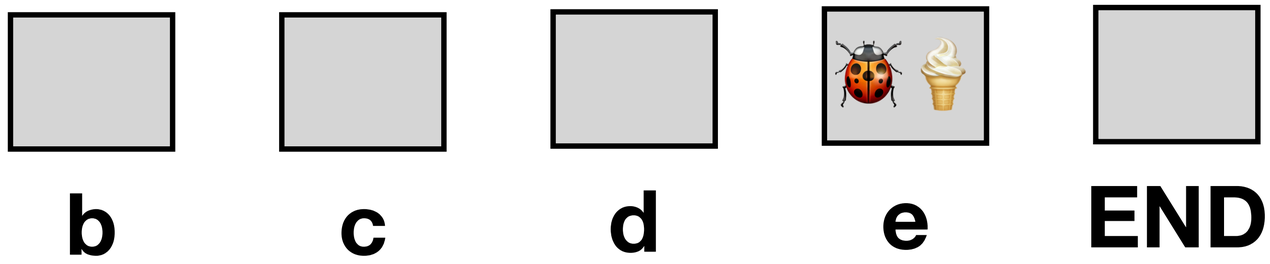
$$\pi_1(a \mid d) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$



$e_1 = d$, right, 1, e

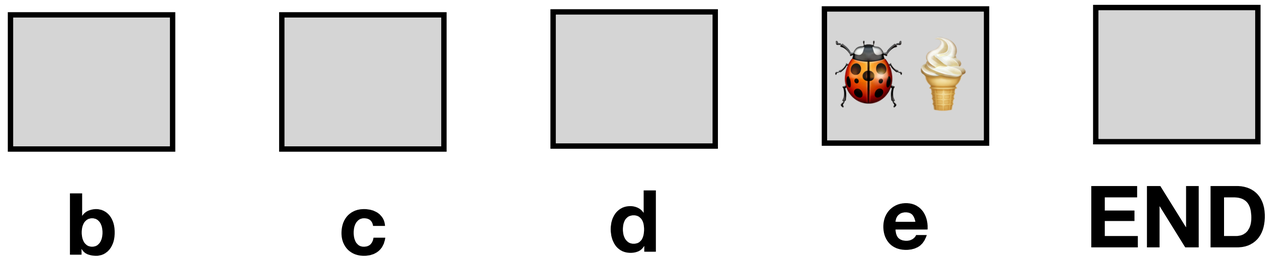
$$\pi_1(a \mid e) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$



$e_1 = d$, right, 1, e, right

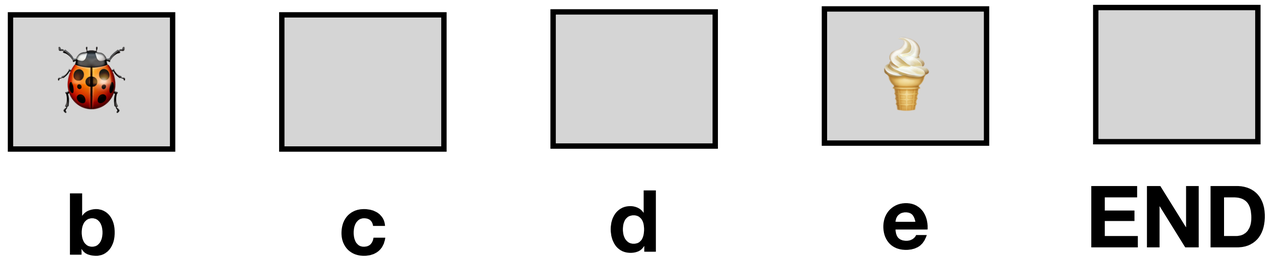
$$\pi_1(a \mid e) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$



$e_1 = d$, right,1,e, right,1,b

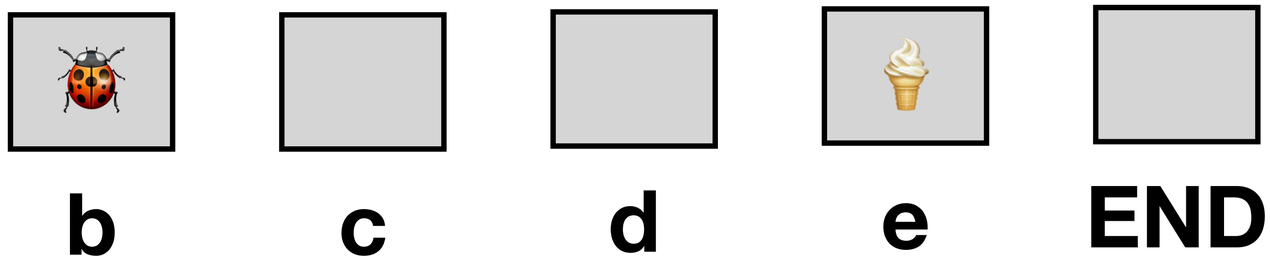
$$\pi_1(a \mid b) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$



$e_1 = d$, right,1,e, right,1,b,eat

$$\pi_1(a \mid b) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Let's run MC On-Policy Iteration on our running example ($\gamma = 0.5$):

$k = 1, \varepsilon = 1$

$e_1 = d$, right,1,e, right,1,b,eat,0,END

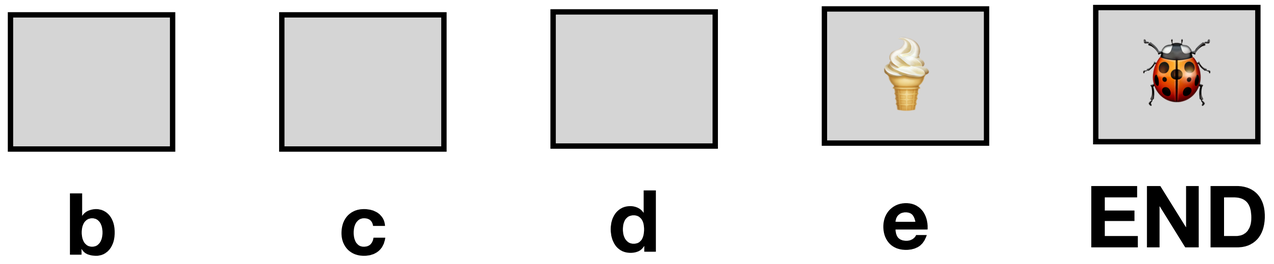
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Episode 1)

Now we use First-Visit MC to update G, N and Q.

$k = 1, \varepsilon = 1$

$e_1 = d$, right,1,e, right,1,b,eat, T



G(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

N(s,a)	*left*	*right*	*eat*
b	0	0	1
c	0	0	0
d	0	1	0
e	0	1	0

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

Running Example (Episode 1)

Now we use First-Visit MC to update G, N and Q.

$k = 1, \varepsilon = 1$

$e_1 = d, \text{right}, 1, \text{e}, \text{right}, 1, b, \text{eat}, T$

G(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

N(s,a)	*left*	*right*	*eat*
b	0	0	1
c	0	0	0
d	0	1	0
e	0	1	0

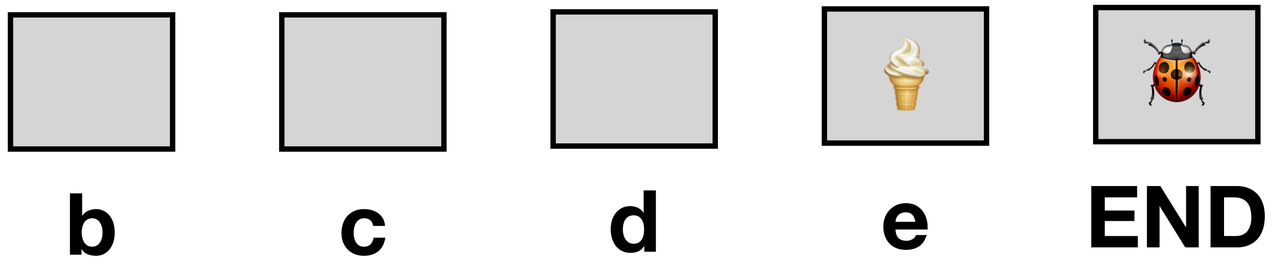
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

Running Example (Episode 1)

Now we use First-Visit MC to update G, N and Q.

$k = 1, \varepsilon = 1$

$e_1 = d, \text{right}, 1, \text{e}, \text{right}, 1, b, \text{eat}, T$



G(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

N(s,a)	*left*	*right*	*eat*
b	0	0	1
c	0	0	0
d	0	1	0
e	0	1	0

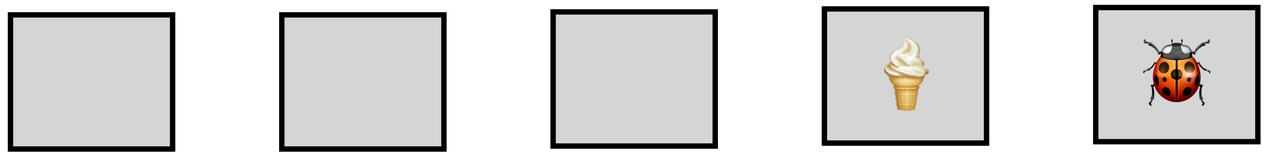
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

Running Example (Episode 1)

Now we use First-Visit MC to update G, N and Q.

$k = 1, \varepsilon = 1$

$e_1 = d, \text{right}, 1, \text{e}, \text{right}, 1, b, \text{eat}, T$



G(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

N(s,a)	*left*	*right*	*eat*
b	0	0	1
c	0	0	0
d	0	1	0
e	0	1	0

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

Running Example (Episode 1)

Now we update the policy π. First, we get the greedy policy w.r.t. $Q(s, a)$

$k = 1, \varepsilon = 1$

b **c** **d** **e** **END**

Let us suppose that if there is tie in $\arg\max_{a \in A}$ then the preference is eat $\prec$ right $\prec$ left (i.e. we prefer left over right and right over eat)

$$\pi_{\text{greedy}}(d) = \pi_{\text{greedy}}(e) = \text{right},$$

$$\pi_{\text{greedy}}(b) = \pi_{\text{greedy}}(c) = \text{left}.$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	1.5	0
e	0	1	0

Running Example (Episode 1)

Now we update the policy π. First, we get the greedy policy w.r.t. $Q(s, a)$

Now, we update $k = 2; \varepsilon = 0.5$

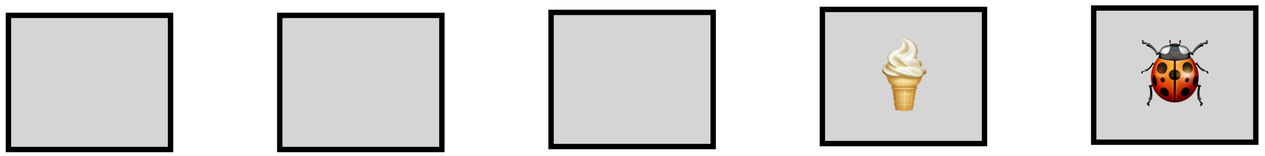
b **c** **d** **e** **END**

The new policy π will be the ε-greedy policy:

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

We then run the next iteration with this new policy π.

Running Example (Episode 1)



As k increases, the algorithm will converge to the optimal policy:

$$\pi(b) = \textbf{left}, \pi(c) = \textbf{left}, \pi(d) = \textbf{right}, \pi(e) = \textbf{eat}$$

GLIE

- We say that an algorithm has the GLIE property (= “greedy in the limit of infinite exploration”), if it satisfies the following two conditions):

- **Definition** (GLIE conditions**):**

  1. If a state $s \in S$ is visited infinitely often, then each action in that state is chosen infinitely often (with probability 1)

  2. In the limit (as t → ∞), the learning policy is greedy with respect to the learned Q-function (with probability 1). By *greedy* we mean (ignoring the possibility of ties in the $\arg\max$ for simplicity) that

$$\pi_{k+1}(a \mid s) = \begin{cases} 1 & \text{for } a = \arg\max_{a \in A} Q_k(s, a), \\ 0 & \text{otherwise.} \end{cases}$$

MC Policy Iteration with $\varepsilon_i = 1/i$ is GLIE

- For a proof, see, e.g. *Singh, S., Jaakkola, T., Littman, M. L., & Szepesvári, C. (2000). Convergence results for single-step on-policy reinforcement-learning algorithms. Machine learning, 38(3), 287-308.*

- The formal proof is a bit tricky…

- **Note:** *There are other seauences of ε_i which guarantee GLIE as well.*

A Theorem (Why GLIE Matters)

- **Theorem**: GLIE Monte-Carlo Control converges to the optimal state-action value function, i.e. $Q_k(s, a) \to Q^*(s, a)$ as $k \to \infty$.

Part 7: SARSA and Q-Learning

General Form of TD-Based Methods

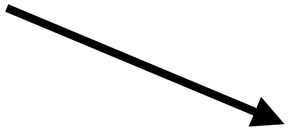
- **Basic idea:**
  - Replace Monte Carlo Policy Evaluation by a temporal-difference method.
  - Still use ε-greedy policies to guarantee that exploration will take place.

Bellman Equations for Q-Function

(Something we skipped when we talked about Q-functions for MDPs but something that will be useful now.)

We have:

$$V^{\pi}(s) = \sum_{a \in A} \pi(a \mid s) \cdot Q^{\pi}(s, a)$$



$$Q^{\pi}(s, a) = R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, a) \cdot V^{\pi}(s')$$

Combining the above:

$$Q^{\pi}(s, a) = R(s, a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s, a) \cdot \sum_{a' \in A} \pi(a' \mid s') \cdot Q^{\pi}(s', a')$$

TD-Target

Bellman for Q-function:

$$Q^{\pi}(s_t, a_t) = R(s_t, a_t) + \gamma \cdot \boxed{\sum_{s_{t+1} \in S} P(s_{t+1} \mid s_t, a_t) \cdot \sum_{a_{t+1} \in A} \pi(a_{t+1} \mid s_{t+1}) \cdot Q^{\pi}(s_{t+1}, a_{t+1})}$$

$$\mathbb{E}[Q^{\pi}(X_{t+1}, A_{t+1}) \mid X_t = s_t, A_t = a_t]$$

Temporal difference update (SARSA)…

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left(\boxed{r_t + \gamma Q(s_{t+1}, a_{t+1})} - Q(s_t, a_t) \right)$$

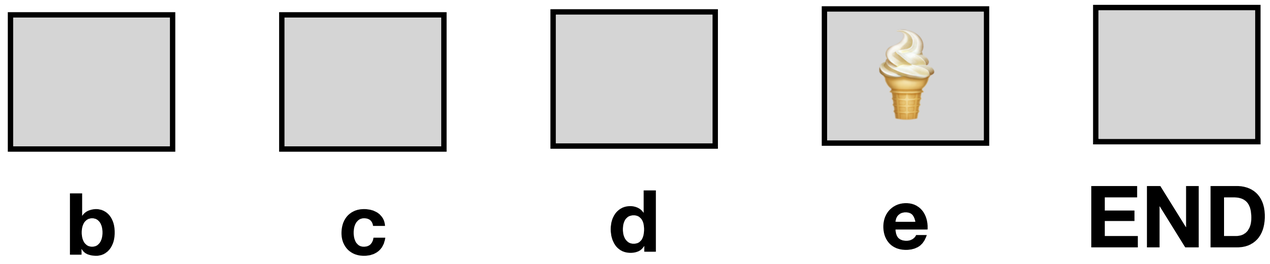
SARSA

1. **Initialize:** set π to be some ε-greedy policy, set $t = 1$, initialize $Q(s, a)$.
2. **Sample** a_1 using the distribution given by π in the state s_1 *(for sampling, we will use the notation* $a_1 \sim \pi(s_1)$*)*.
3. **While** s_t is not a terminal state:
    1. **Take** action a_t and observe r_t, s_{t+1}.
    2. **Sample** $a_{t+1} \sim \pi(s_{t+1})$ and store it for the next iteration.
    3. $Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right)$
    4. $\pi := \varepsilon$-greedy(Q)
    5. Set $t := t + 1$. Update ε, α */* see next slides */*

Running Example (Initialization)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.
$t = 1, \varepsilon = 1, \alpha = 0.1$

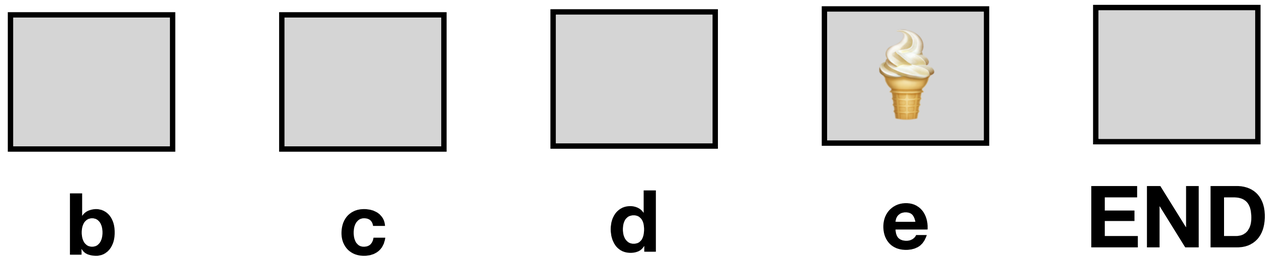


Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Initialization)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.
$t = 1, \varepsilon = 1, \alpha = 0.1$

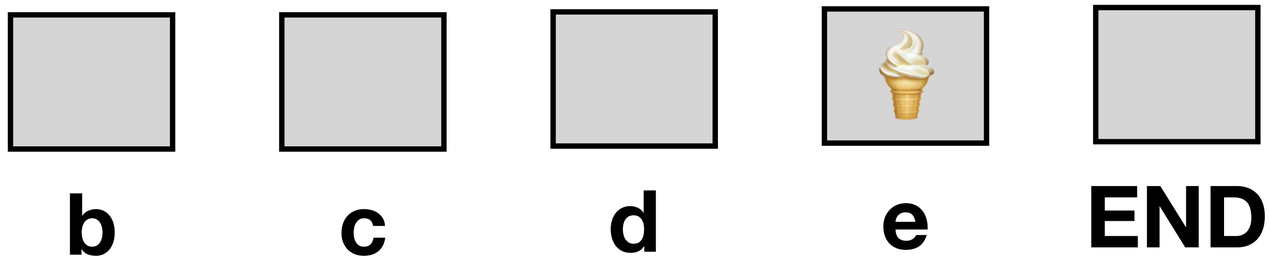


Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Initialization)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.
$t = 1, \varepsilon = 1, \alpha = 0.1$



Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example (Initialization)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$

World samples the state $s_1 = d$

$s_1 = d$

b **c** **d** **e** **END**

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

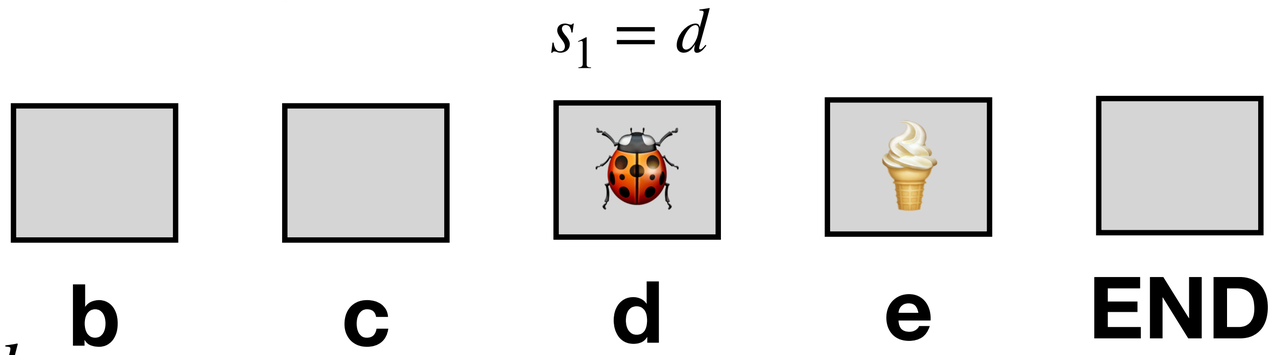
Running Example (Initialization)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$

World samples the state $s_1 = d$



We sample a_1 (we do not take it yet)

$$a_1 \sim \pi_1(a \mid d) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

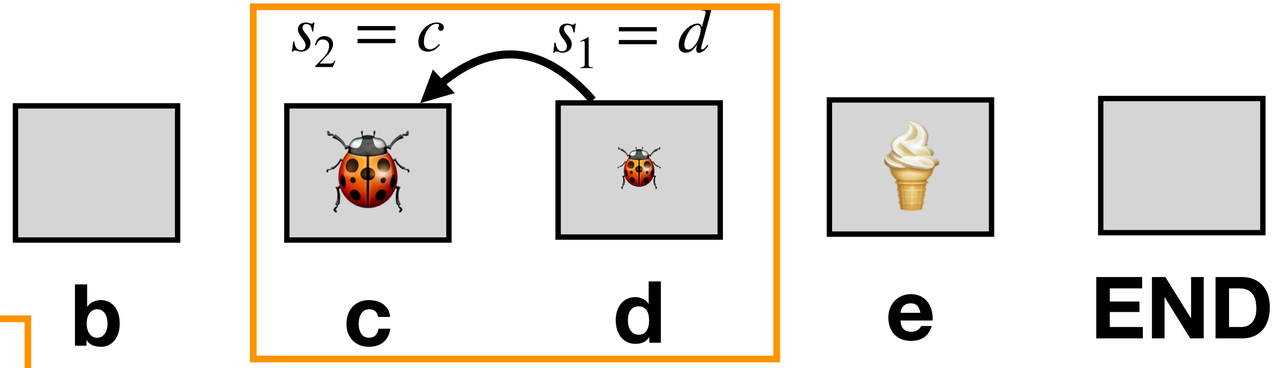
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example ($t = 1$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We take the action $a_1 =$ left

We observe: $r_1 = 1$ **and** $s_2 = c$

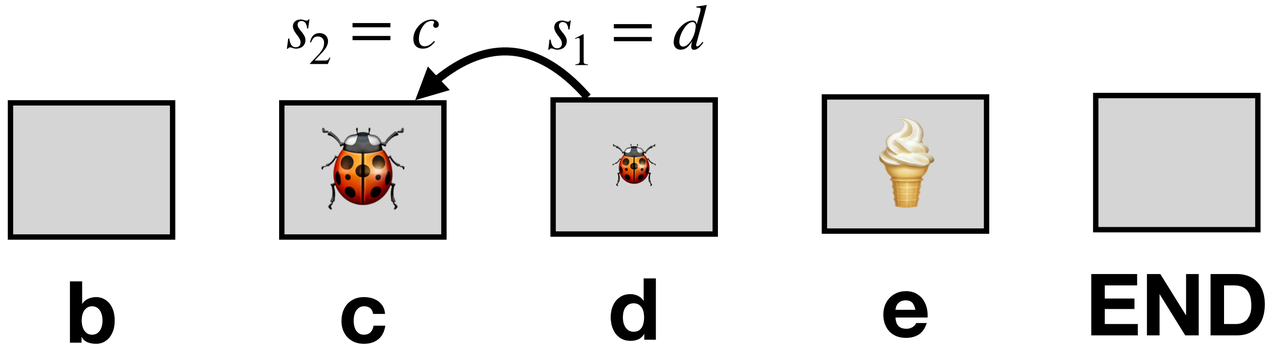
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example ($t = 1$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We have: $r_1 = 1$ **and** $s_2 = c$

We sample a_2 (we are not taking it yet)

$$a_2 \sim \pi_1(a \mid c) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Say, it is $a_2 = \text{left}$.

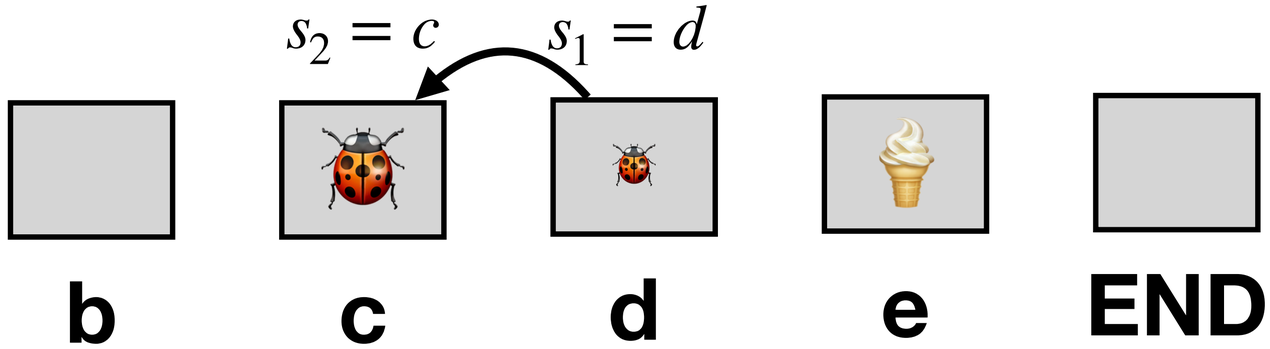
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example ($t = 1$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We have: $r_1 = 1$ **and** $s_2 = c$

We sample a_2 (we are not taking it yet)

$$a_2 \sim \pi_1(a \mid c) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Say, it is $a_2 = \text{left}$.

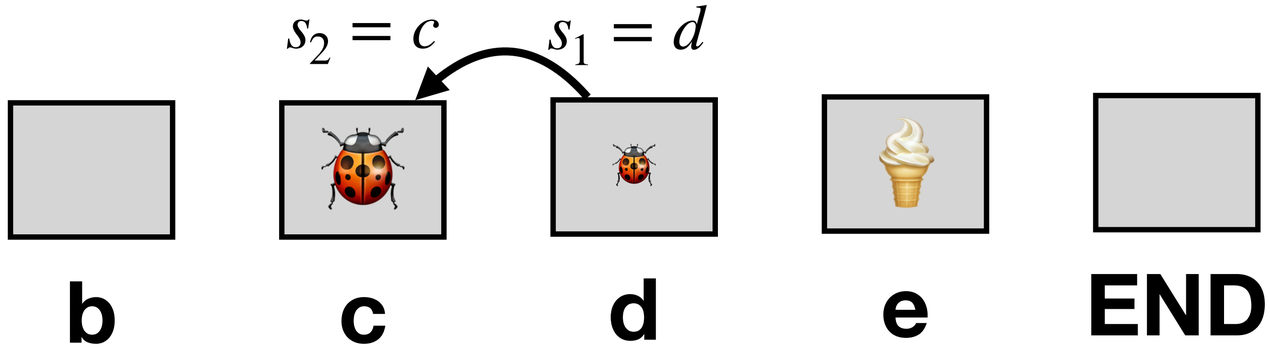
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example ($t = 1$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We have: $r_1 = 1$ **and** $s_2 = c$

We sample a_2 (we are not taking it yet)

$$a_2 \sim \pi_1(a \mid c) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Say, it is $a_2 = \text{left}$.

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

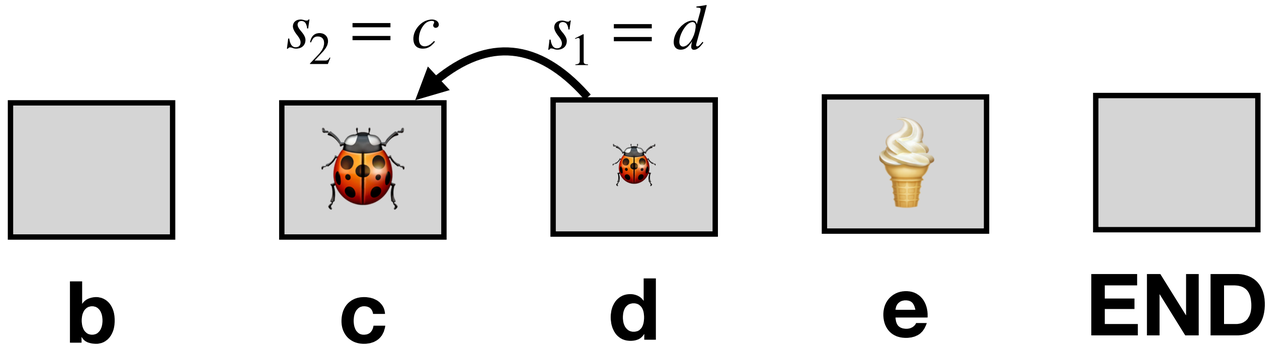
Running Example ($t = 1$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$

We have: $r_1 = 1$ **and** $s_2 = c$



We now update the Q-function:

$$Q(d, \text{left}) := 0 + 0.1\left(1 + 0.5 \cdot 0 - 0\right) = 0.1$$

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha\left(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right)$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	**0.1**	0	0
e	0	0	0

Running Example ($t = 2$)

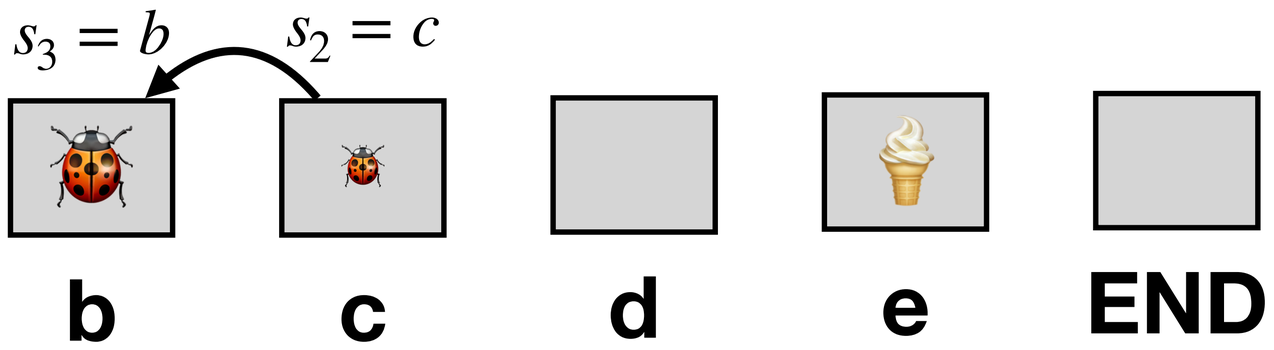
Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$

We take the action $a_2 =$ left

We observe: $r_2 = 1$ **and** $s_3 = b$



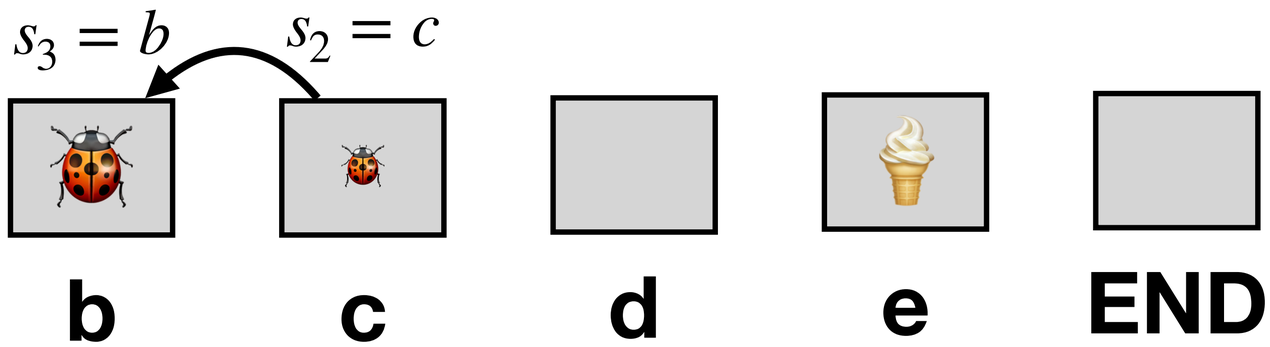
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 2$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$



We take the action $a_2 = \text{left}$

We observe: $r_2 = 1$ **and** $s_3 = b$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 2$)

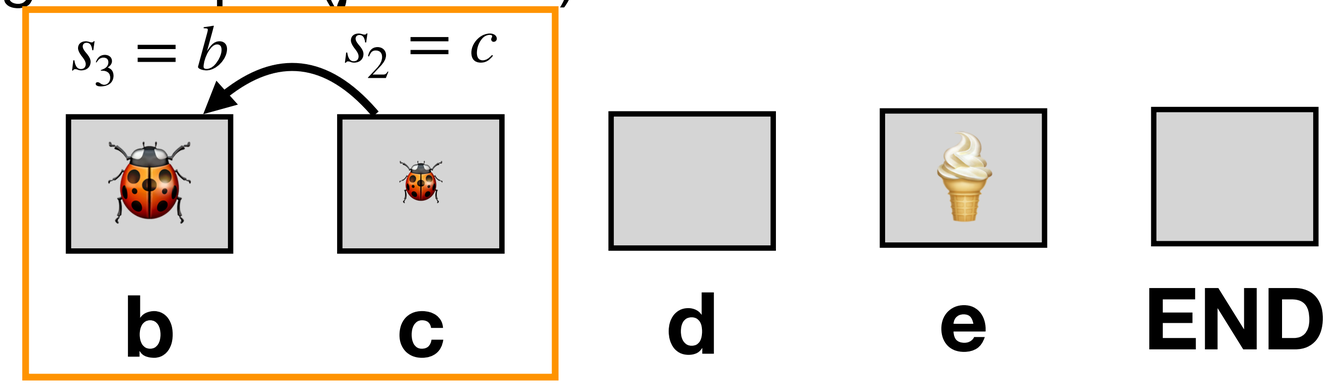
Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$

We take the action $a_2 =$ left

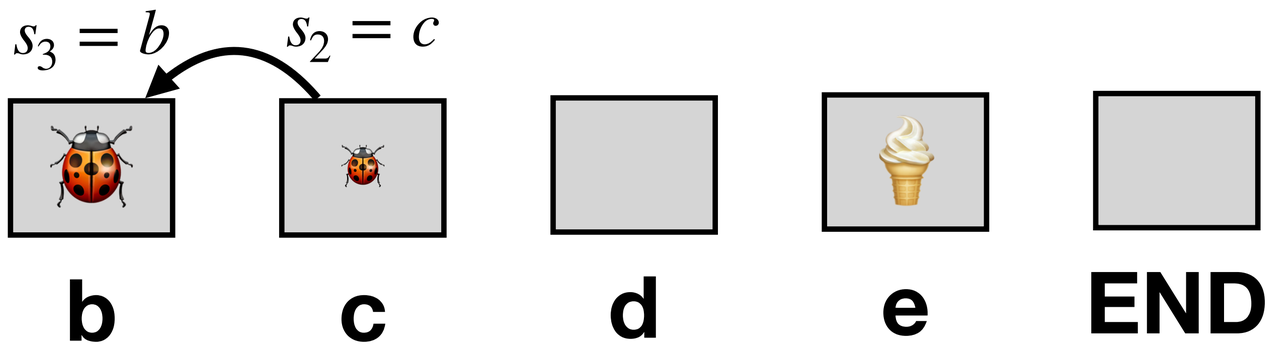
We observe: $r_2 = 1$ **and** $s_3 = b$



Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 2$)

Let's run SARSA on our running example ($\gamma = 0.5$):



We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$

We have: $r_2 = 1$ **and** $s_3 = b$

We sample a_3 (we are not taking it yet)

$$\pi_1(a \mid b) = \begin{cases} 1 - 0.5 + 1/6 = 2/3 & a = \text{left} \\ 1/6 & a = \text{right} \\ 1/6 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

What happened here: Even though we did not update the estimates of the Q-function for the state c, the policy changed. Recall that we break ties (we have the preference eat $\prec$ right $\prec$ left and recall how we define greedy and ε-greedy policies.

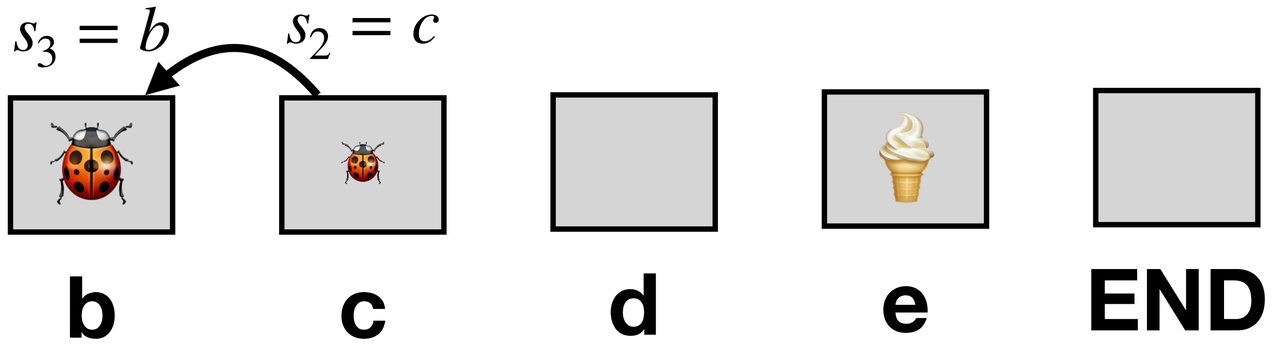
Say, it is $a_3 = \text{right}$.

Running Example ($t = 2$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$



We have: $r_2 = 1$ **and** $s_3 = b$

We sample a_3 (we are not taking it yet)

$$\pi_1(a \mid b) = \begin{cases} 1 - 0.5 + 1/6 = 2/3 & a = \text{left} \\ 1/6 & a = \text{right} \\ 1/6 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

What happened here: Even though we did not update the estimates of the Q-function for the state c, the policy changed. Recall that we break ties (we have the preference eat $\prec$ right $\prec$ left and recall how we define greedy and ε-greedy policies.

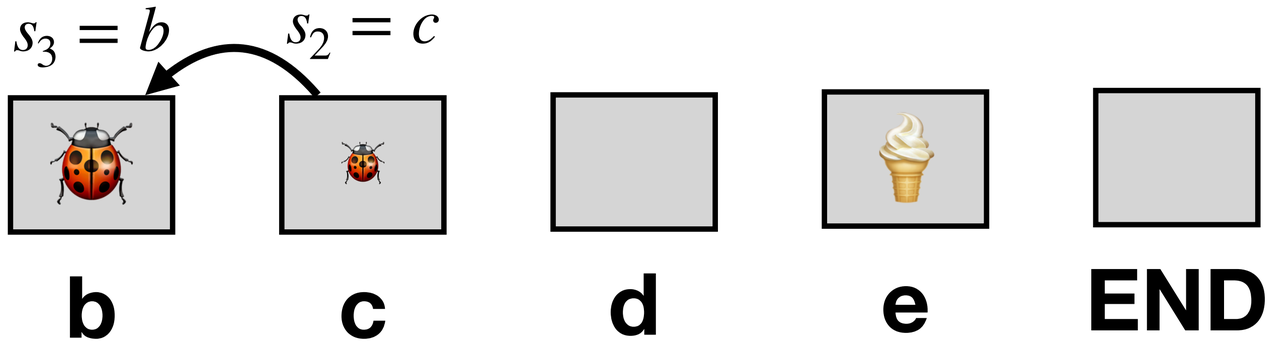
Say, it is $a_3 = \text{right}$.

Running Example ($t = 2$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$



We have: $r_2 = 1$ **and** $s_3 = b$

We sample a_3 (we are not taking it yet)

$$\pi_1(a \mid b) = \begin{cases} 1 - 0.5 + 1/6 = 2/3 & a = \text{left} \\ 1/6 & a = \text{right} \\ 1/6 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

What happened here: Even though we did not update the estimates of the Q-function for the state c, the policy changed. Recall that we break ties (we have the preference eat $\prec$ right $\prec$ left and recall how we define greedy and ε-greedy policies.

Say, it is $a_3 = \text{right}$.

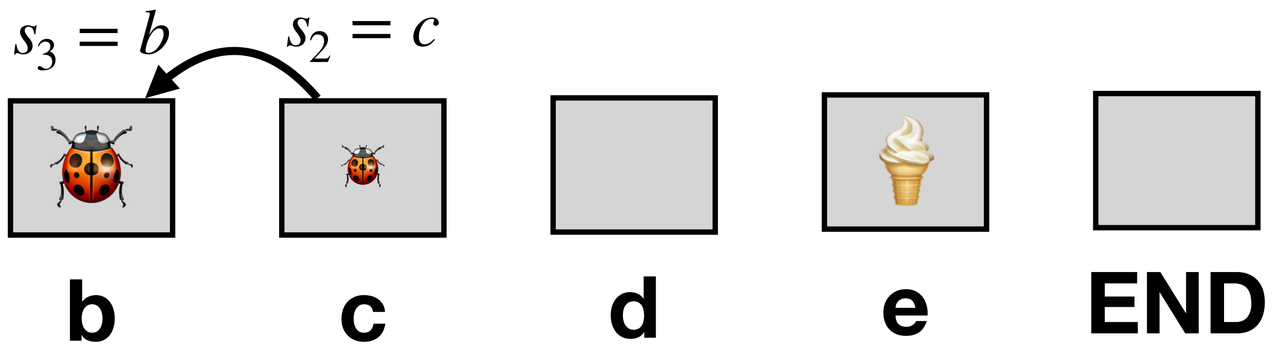
Running Example ($t = 2$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 2,\ \varepsilon = 0.5,\ \alpha = 0.1$

We have: $r_2 = 1$ **and** $s_3 = b$



We now update the Q-function:

$$Q(c, \text{left}) := 0 + 0.1\left(1 + 0.5 \cdot 0 - 0\right) = 0.1$$

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha\left(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right)$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	**0.1**	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 3$)

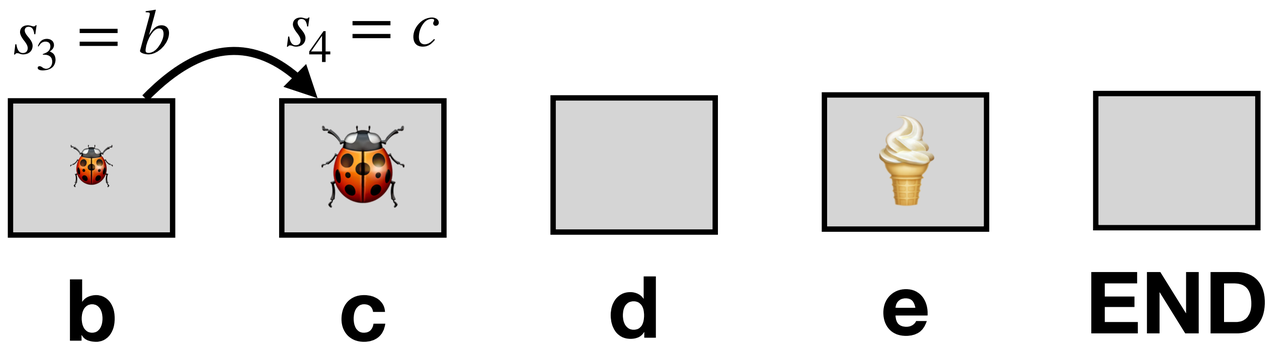
Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$

We take the action $a_3 = \text{right}$

We observe: $r_3 = 1$ **and** $s_4 = c$**.**



Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0.1	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 3$)

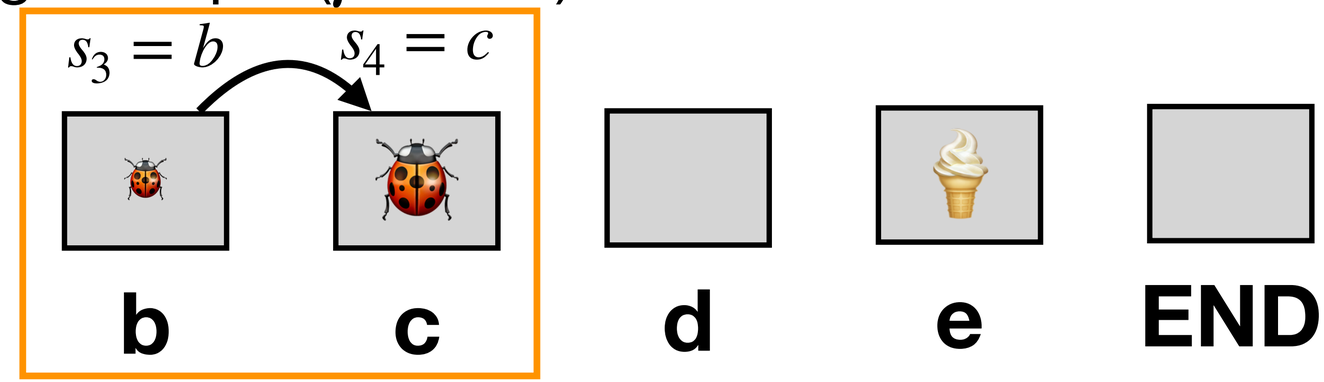
Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$

We take the action $a_3 = \text{right}$

We observe: $r_3 = 1$ **and** $s_4 = c$**.**



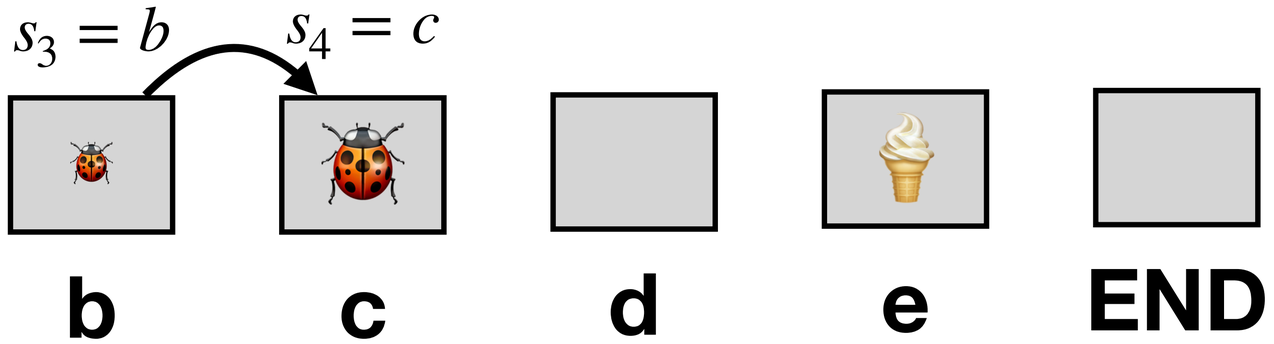
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0.1	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 3$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We have: $r_3 = 1$ **and** $s_4 = c$

We sample a_4 (we are not taking it yet)

$$\pi_1(a \mid c) = \begin{cases} 7/9 & a = \text{left} \\ 1/9 & a = \text{right} \\ 1/9 & a = \text{eat} \end{cases}$$

Say, it is $a_4 = \text{left}$.

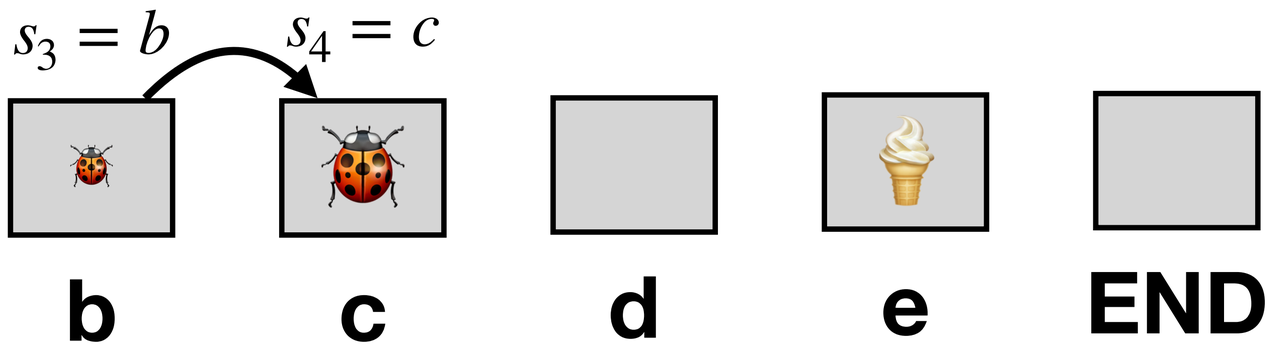
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0.1	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 3$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We have: $r_3 = 1$ **and** $s_4 = c$

We sample a_4 (we are not taking it yet)

$$\pi_1(a \mid c) = \begin{cases} 7/9 & a = \text{left} \\ 1/9 & a = \text{right} \\ 1/9 & a = \text{eat} \end{cases}$$

Say, it is $a_4 = \text{left}$.

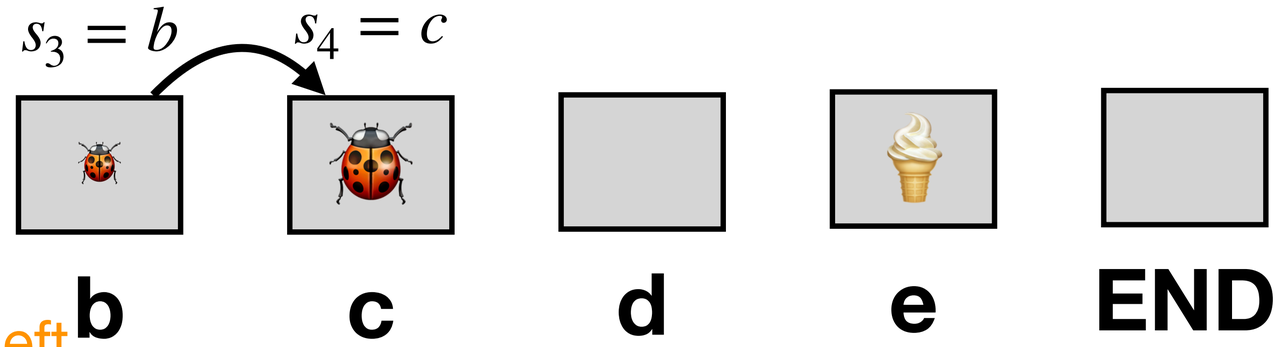
Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0.1	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 3$)

Let's run SARSA on our running example ($\gamma = 0.5$):

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$



We have: $r_3 = 1$ **and** $s_4 = c$, $a_4 = \text{left}$

We now update the Q-function:

$$Q(b, \text{right}) := 0 + 0.1 \cdot (1 + 0.5 \cdot 0.1 - 0.1) = 0.095$$

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right)$$

Q(s,a)	*left*	*right*	*eat*
b	0	0.095	0
c	0.1	0	0
d	0.1	0	0
e	0	0	0

AND SO ON….

Note: Breaking Ties

It is usually suggested as a good idea to break ties randomly.

Indeed, as we saw in our example, without tie breaking our Q-values were prefering some actions in states we have not even visited yet, just because of the arbitrary tie breaking.

Let us rerun the example where we define the greedy policy with random tie breaking and ε-greedy policy as:

$$\pi_{\varepsilon}(a \mid s) = (1 - \varepsilon) \cdot \pi_{\text{greedy}}(a \mid s) + \frac{\varepsilon}{|A|}.$$

Note: We will not be showing all details of the updates in the next slides (that would be redundant to what we already saw). Focus mostly on the ε-policies.

Running Example (Initialization)

We will use $\varepsilon_t = 1/t$.

$t = 1, \varepsilon = 1, \alpha = 0.1$

b c d e END

With random tie-breaking:

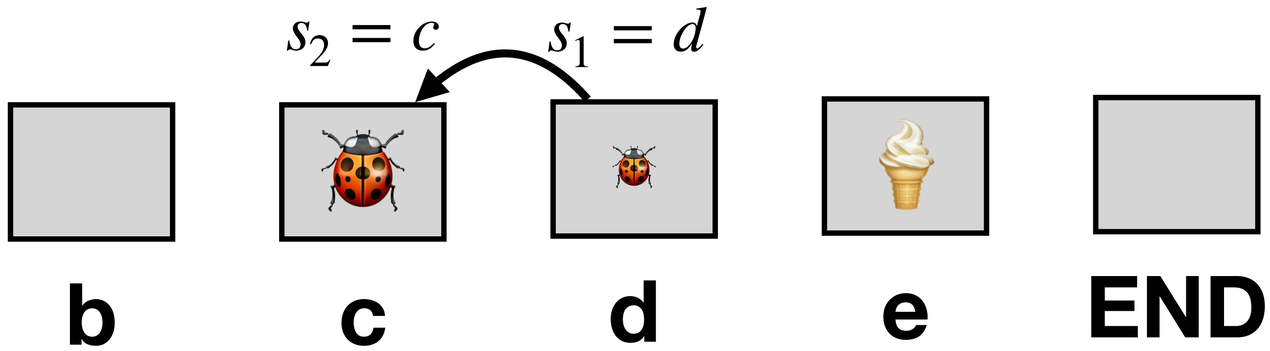
$$a_1 \sim \pi(a \mid d) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Without random tie-breaking:

$$a_1 \sim \pi(a \mid d) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example ($t = 1$)



With random tie-breaking:

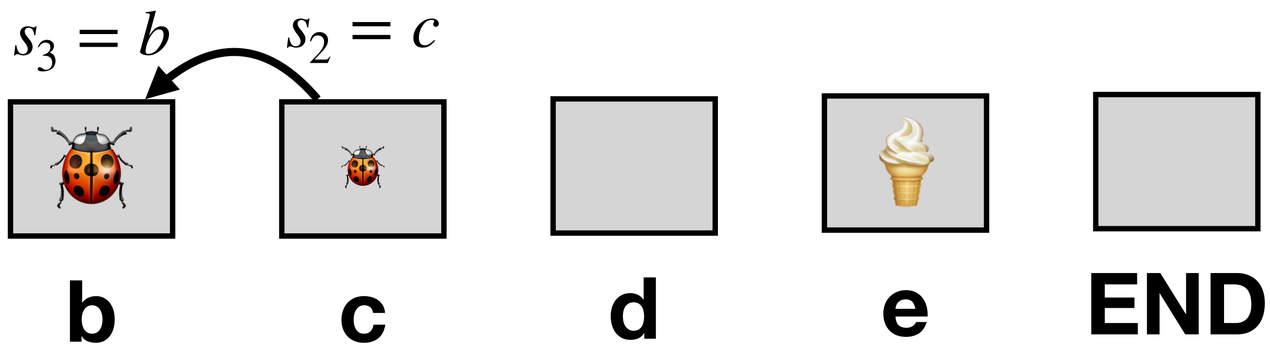
$$a_2 \sim \pi(a \mid c) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Without random tie-breaking:

$$a_2 \sim \pi(a \mid c) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0	0	0
e	0	0	0

Running Example ($t = 2$)



With random tie-breaking:

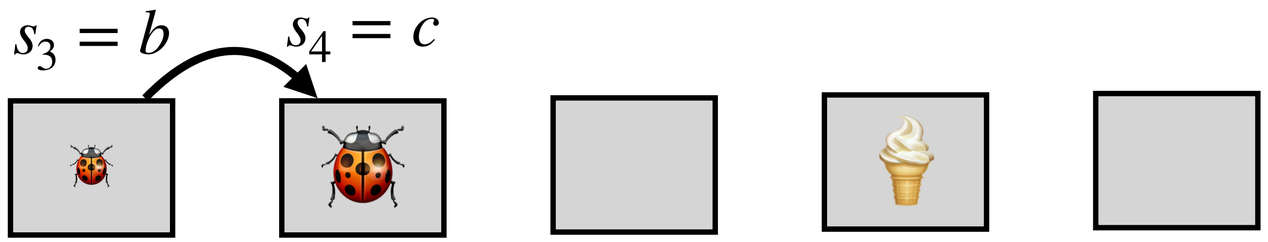
$$a_3 \sim \pi(a \mid b) = \begin{cases} 1/3 & a = \text{left} \\ 1/3 & a = \text{right} \\ 1/3 & a = \text{eat} \end{cases}$$

Without random tie-breaking:

$$a_3 \sim \pi(a \mid b) = \begin{cases} 2/3 & a = \text{left} \\ 1/6 & a = \text{right} \\ 1/6 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	0	0	0
d	0.1	0	0
e	0	0	0

Running Example ($t = 3$)



With random tie-breaking:

$$a_4 \sim \pi(a \mid c) = \begin{cases} 7/9 & a = \text{left} \\ 1/9 & a = \text{right} \\ 1/9 & a = \text{eat} \end{cases}$$

Without random tie-breaking:

$$a_4 \sim \pi(a \mid c) = \begin{cases} 7/9 & a = \text{left} \\ 1/9 & a = \text{right} \\ 1/9 & a = \text{eat} \end{cases}$$

Q(s,a)	*left*	*right*	*eat*
b	0	0	0
c	**0.1**	0	0
d	0.1	0	0
e	0	0	0

AND SO ON….

Note: Optimistic Initialization

What happens if we initialize the Q values differently?

For instance, what would happen if we started with:

Q(s,a)	*left*	*right*	*eat*
b	5	5	5
c	5	5	5
d	5	5	5
e	5	5	5

Answer: The agent would be “exploring” more than with the initialization we used.

This is a general property. If you want to promote exploration, initialize higher estimate of the Q function.

Convergence (SARSA)

- SARSA converges to the optimal state-value function Q^* if the following conditions are satisfied:

    1. The sequence of policies π_t satisfies the GLIE conditions (enough to have $\varepsilon_t = 1/t$).

    2. Step-sizes satisfy the Robbins-Monro conditions:

$$\sum_{t=1}^{\infty} \alpha_t = \infty,$$

$$\sum_{t=1}^{\infty} \alpha_t^2 < \infty.$$

Note: Why “SARSA”?

Why the name? Because of the update rule

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left(r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right)$$

which uses the tuple $s_t, a_t, r_t, s_{t+1}, a_{t+1}$ ~ s a r s a.

Q-Learning (1/2)

- The **Optimal Bellman Equation** (we have not talked about it yet but it is similar to what we already saw)**:**

$$Q^*(s,a) = R(s,a) + \gamma \sum_{s_{t+1} \in S} P(s_{t+1} \mid s_1, a_t) \cdot \max_{a_{t+1} \in A} Q^*(s_{t+1}, a_{t+1}).$$

$$\mathbb{E}\left[\max_{a_{t+1} \in A} Q^*(X_{t+1}, a_{t+1}) \,\middle|\, X_t = s_t, A_t = a_t \right]$$

- Q-Learning update rule:

$$Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left(r_t + \gamma \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a_t) \right)$$

Q-Learning (2/2)

1. **Initialize:** set π to be some ε-greedy policy, set $t = 1$
2. **Observe the initial state** s_1
3. **While** s_t is not a terminal state:
    1. **Take** action $a_t \sim \pi(s_t)$ and observe r_t, s_{t+1}.
    2. $Q(s_t, a_t) := Q(s_t, a_t) + \alpha \left(r_t + \gamma \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a_t) \right)$
    3. $\pi := \varepsilon$-greedy(Q)
    4. Set $t := t + 1$. Update ε, α */* see next slides */*

Convergence (Q-Learning)

- For convergence of the state-value Q-function, we need only the Robbins-Monro conditions + every state-action pair needs to be visited infinitely often (with probability 1).

- For convergence of the policy to the optimal policy, we need GLIE (i.e. it needs to also be greedy in the limit…).

On-Policy and Off-Policy Methods

- **On-policy methods**: samples must be from the policy that we are learning. **Example:** SARSA, MC Policy Iteration.

- **Off-policy methods**: samples do not have to be from the policy that we are learning. **Example:** Q-Learning.

END OF SLIDES