2. Khái niệm cơ bản về Expected SARSA

- **SARSA (State-Action-Reward-State-Action)** là một thuật toán **on-policy TD Control**, nghĩa là agent vừa **thực thi** chính sách π để tương tác, vừa dùng chính các trải nghiệm đó để cập nhật Q^{π}.
- **Expected SARSA** là một biến thể của SARSA, thay vì chỉ sử dụng **giá trị** $Q(s_{t+1}, a_{t+1})$ của một hành động duy nhất a_{t+1} đã chọn, nó **tính trung bình (expected value)** của $Q(s_{t+1}, \cdot)$ trên toàn bộ tập hành động có khả năng được chọn theo chính sách hiện tại.

    Lý do: Bằng cách xét cả phân phối xác suất của các hành động tiềm năng, Expected SARSA giảm biến thiên (variance) của cập nhật so với SARSA, làm cho quá trình hội tụ mượt mà hơn.

---

3. Quy tắc cập nhật (Update Rule)

Giả sử tại thời điểm t agent:

- Ở trạng thái $S_t = s$, thực thi hành động $A_t = a$,
- Quan sát reward $R_{t+1} = r$ và chuyển sang trạng thái kế tiếp $S_{t+1} = s'$.

3.1. Công thức SARSA truyền thống

SARSA on-policy cập nhật theo:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)]$$

với $a' \sim \pi(s')$ là hành động thực sự chọn ở bước kế theo policy.

3.2. Công thức Expected SARSA

Expected SARSA thay thế $Q(s',a')$ bằng **kỳ vọng** theo phân phối π hiện tại:

$$\mathrm{E}_{a' \sim \pi(\cdot|s')}[Q(s',a')] = \sum_{a'' \in A(s')} \pi(a'' \mid s')\, Q(s',a'').$$

Do đó, quy tắc cập nhật trở thành:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \underbrace{\sum_{a''} \pi(a'' \mid s')\, Q(s',a'')}_{\text{Expected Q at } s'} - Q(s,a)].$$

Chú ý về $\pi(a'' \mid s')$:

- Nếu policy đang dùng là **ε-greedy**, thì

$$\pi(a'' \mid s') = \begin{cases} 1 - \varepsilon + \frac{\varepsilon}{|A|}, & \text{nếu } a'' = \arg\max_a Q(s', a), \\ \frac{\varepsilon}{|A|}, & \text{ngược lại.} \end{cases}$$

---

4. Pseudo-code của Expected SARSA

plaintext Sao chép Chỉnh sửa

```
Input: learning rate α, discount γ, exploration ε Initialize Q(s,a) arbitrarily for all s,a for episode = 1 to M do Initialize state s loop for each step of episode: • Chọn action a theo ε-greedy dựa trên Q(s,·) • Thực thi a, quan sát r và next state s' • Tính ExpectedQ = ∑_{a''} π(a''|s') · Q(s',a'') • Cập nhật: Q(s,a) ← Q(s,a) + α [r + γ·ExpectedQ − Q(s,a)] • s ← s' until s is terminal end for
```

---

5. So sánh Expected SARSA vs SARSA

Đặc điểm	SARSA	Expected SARSA
Nguồn cập nhật	Dùng đúng Q(s′, a′) của hành động được chọn	Dùng trung bình kỳ vọng ∑ π(a″
Biến thiên (Variance)	Cao hơn (dễ dao động nếu a′ mang giá trị Q thay đổi)	Thấp hơn (làm mịn các cập nhật)
Tính on-policy	Cả chọn và cập nhật đều theo policy hiện tại	Vẫn on-policy (dùng π để tính expected), nhưng ổn định hơn
Khám phá – Khai thác	ε-greedy: có thể gây bước ngẫu nhiên lớn khiến cập nhật ồn	Vẫn ε-greedy, nhưng xem xét xác suất của all actions
Hội tụ	Hội tụ về Qπ của chính π	Hội tụ nhanh và ổn định hơn về Qπ

---

6. Expected SARSA so với Q-Learning

Đặc điểm	Q-Learning (off-policy)	Expected SARSA (on-policy)
Target khi cập nhật	$r + \gamma \cdot \max_{a'} Q(s', a')$	(r + γ·\sum_{a''} π(a''
Chính sách hành động	ε-greedy (behavior) nhưng học Q* (target luôn greedy)	ε-greedy (behavior = target); cập nhật dựa trên π
Biến thiên	Cao (do max operator)	Thấp hơn (do trung bình expected)
Tính ổn định	Có thể dao động nếu max bên ngoài distribution	Ổn định hơn, hội tụ mượt mà

Expected SARSA nằm giữa SARSA và Q-Learning:

- **Giữ phần on-policy của SARSA.**
- **Giảm variance nhờ tính trung bình như Q-Learning nhưng không "tham lam" hoàn toàn như phép max.**

---

7. Ví dụ minh họa: Grid World đơn giản

Giả sử môi trường 1-D có 5 ô: 0 (start), 4 (goal, reward +1), các ô khác reward 0. Agent có hai action: "Lên" (→) hoặc "Xuống" (←).

1. **Khởi tạo**:
   - $Q(s, a) = 0$ cho mọi $s \in \{0, 1, 2, 3, 4\}, a \in \{\leftarrow, \rightarrow\}$.
   - $\alpha = 0.1,\ \gamma = 0.9,\ \varepsilon = 0.1.$
2. **Bước ví dụ**: Agent ở $s = 2$.
   - Chọn action "→" với xác suất 0.9 (greedy) hoặc "←" với xác suất 0.1 (explore).
   - Giả sử chọn "→", đến $s' = 3$, reward $r = 0$.
   - Tại $s' = 3$, policy ε-greedy có xác suất 0.9 chọn "→" và 0.1 chọn "←".
     - Giả sử Q(3,→)=0.5, Q(3,←)=0.2.
     - Thì

$$\text{ExpectedQ} = 0.9 \cdot 0.5 + 0.1 \cdot 0.2 = 0.45 + 0.02 = 0.47.$$

   - Cập nhật:

$$Q(2, \rightarrow) \leftarrow 0 + 0.1[0 + 0.9 \cdot 0.47 - 0] = 0.1 \times 0.423 = 0.0423.$$

3. **Lặp lại** cho đến khi agent thường xuyên đi từ 0 → 4, lúc đó Q-table sẽ thể hiện rõ "đường tắt" và policy greedy tương ứng.

---

8. Lợi ích chính của Expected SARSA

- **Giảm độ dao động** (variance) trong cập nhật so với SARSA.
- **Ổn định hội tụ** hơn, nhất là khi policy có tính ngẫu nhiên cao.
- **Tổng quát hoá Q-Learning**: nếu policy π luôn chọn greedy (ε→0), Expected SARSA suy giản về Q-Learning.

Kết luận

Expected SARSA là một bước cải tiến quan trọng khi bạn muốn vừa giữ tính on-policy của SARSA, vừa giảm thiểu biến thiên cập nhật bằng cách xem xét giá trị trung bình của tất cả hành động trong bước kế. Điều này giúp quá trình học nhanh hơn, ổn định hơn và dễ dàng mở rộng sang những môi trường phức tạp, nơi variance cao có thể gây ra dao động lớn hoặc hội tụ chậm.