Reinforcement Learning
Tutorial for Lecture 4

Georg Martius

Distributed Intelligence / Autonomous Learning Group, Uni Tübingen, Germany

November 5, 2024

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Greedy policy

We have a given MDP and assume we have estimated the Value function $V(s)$.
How do I get the greedy policy?

Greedy policy

We have a given MDP and assume we have estimated the Value function $V(s)$.
How do I get the greedy policy?

$$\pi'(s) = \arg\max_{a \in \mathcal{A}} \mathcal{R}_s^a + \mathcal{P}_{ss'}^a V(s')$$

Yes, we need Reward and Transition Kernel! So we need access to the MDP.

Greedy policy

We have a given MDP and assume we have estimated the Value function $V(s)$. How do I get the greedy policy?

$$\pi'(s) = \arg\max_{a \in \mathcal{A}} \mathcal{R}_s^a + \mathcal{P}_{ss'}^a V(s')$$

Yes, we need Reward and Transition Kernel! So we need access to the MDP.

How do we solve this problem in model-free control?

Greedy policy

We have a given MDP and assume we have estimated the Value function $V(s)$.
How do I get the greedy policy?

$$\pi'(s) = \underset{a \in \mathcal{A}}{\arg\max}\, \mathcal{R}_s^a + \mathcal{P}_{ss'}^a V(s')$$

Yes, we need Reward and Transition Kernel! So we need access to the MDP.

How do we solve this problem in model-free control?
Learn an action-value function!

Action-value function

$q_\pi(s, a)$: value of performing action a in state s and then following the policy
$Q(s, a)$ is an estimate of $q_\pi(s, a)$

Write down the greedy policy using Q:

Greedy policy

We have a given MDP and assume we have estimated the Value function $V(s)$.
How do I get the greedy policy?

$$\pi'(s) = \arg\max_{a \in \mathcal{A}} \mathcal{R}_s^a + \mathcal{P}_{ss'}^a V(s')$$

Yes, we need Reward and Transition Kernel! So we need access to the MDP.

How do we solve this problem in model-free control?
Learn an action-value function!

Action-value function

$q_\pi(s, a)$: value of performing action a in state s and then following the policy
$Q(s, a)$ is an estimate of $q_\pi(s, a)$

Write down the greedy policy using Q:

$$\pi'(s) = \arg\max_{a \in \mathcal{A}} Q(s, a)$$

ϵ-Greedy exploration

Explain to you neighbor what ϵ-Greedy exploration is.

ϵ-Greedy exploration

Explain to you neighbor what ϵ-Greedy exploration is.

- All m actions are tried with non-zero probability
- With probability $1-\epsilon$ choose the greedy action
- With probability ϵ choose an action at random

ϵ-Greedy exploration

Explain to you neighbor what ϵ-Greedy exploration is.

- All m actions are tried with non-zero probability
- With probability $1 - \epsilon$ choose the greedy action
- With probability ϵ choose an action at random

Write the greedy (no exploration) policy $\pi(a \mid s)$ with respect to $Q(s, a)$ (now it is a distribution so it should return probability for action a):

ϵ-Greedy exploration

Explain to you neighbor what ϵ-Greedy exploration is.

- All m actions are tried with non-zero probability
- With probability $1-\epsilon$ choose the greedy action
- With probability ϵ choose an action at random

Write the greedy (no exploration) policy $\pi(a \mid s)$ with respect to $Q(s,a)$ (now it is a distribution so it should return probability for action a):

$$\pi(a \mid s) = \begin{cases} 1 & \text{if } a^* = \arg\max_{a \in \mathcal{A}} Q(s,a) \\ 0 & \text{otherwise} \end{cases}$$

ϵ-Greedy exploration

Explain to you neighbor what ϵ-Greedy exploration is.

- All m actions are tried with non-zero probability
- With probability $1 - \epsilon$ choose the greedy action
- With probability ϵ choose an action at random

Write the greedy (no exploration) policy $\pi(a \mid s)$ with respect to $Q(s, a)$ (now it is a distribution so it should return probability for action a):

$$\pi(a \mid s) = \begin{cases} 1 & \text{if } a^* = \arg\max_{a \in \mathcal{A}} Q(s, a) \\ 0 & \text{otherwise} \end{cases}$$

Try to write down the ϵ-greedy policy $\pi(a \mid s)$ with respect to $Q(s, a)$:

ϵ-Greedy exploration

Explain to you neighbor what ϵ-Greedy exploration is.

- All m actions are tried with non-zero probability
- With probability $1 - \epsilon$ choose the greedy action
- With probability ϵ choose an action at random

Write the greedy (no exploration) policy $\pi(a \mid s)$ with respect to $Q(s, a)$ (now it is a distribution so it should return probability for action a):

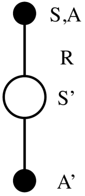
$$\pi(a \mid s) = \begin{cases} 1 & \text{if } a^* = \arg\max_{a \in \mathcal{A}} Q(s, a) \\ 0 & \text{otherwise} \end{cases}$$

Try to write down the ϵ-greedy policy $\pi(a \mid s)$ with respect to $Q(s, a)$:

$$\pi(a \mid s) = \begin{cases} \frac{\epsilon}{m} + 1 - \epsilon & \text{if } a^* = \arg\max_{a \in \mathcal{A}} Q(s, a) \\ \frac{\epsilon}{m} & \text{otherwise} \end{cases}$$

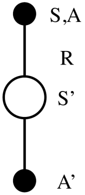
SARSA

Simplest TD Q update formula is the **S A R S A** update. It is "on-policy".
Can you write down the update for Q?



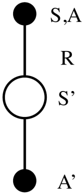
SARSA

Simplest TD Q update formula is the **S A R S A** update. It is "on-policy".
Can you write down the update for Q?



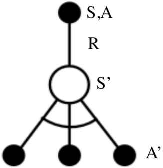
$$Q(S, A) \leftarrow Q(S, A) + \alpha \left(R + \gamma Q(S', A') - Q(S, A) \right)$$

Simplest TD Q update formula is the **S A R S A** update. It is "on-policy". Can you write down the update for Q?



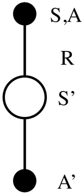
$$Q(S,A) \leftarrow Q(S,A) + \alpha\,(R + \gamma Q(S',A') - Q(S,A))$$

Can you write the **Q-learning** update for Q. It is called off-policy.



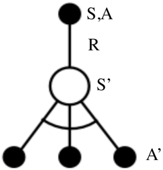
Simplest TD Q update formula is the **S A R S A** update. It is "on-policy". Can you write down the update for Q?

$$Q(S,A) \leftarrow Q(S,A) + \alpha\left(R + \gamma Q(S',A') - Q(S,A)\right)$$



Can you write the **Q-learning** update for Q. It is called off-policy.

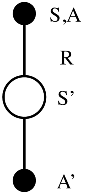
$$Q(S,A) \leftarrow Q(S,A) + \alpha\left(R + \gamma \max_{a'} Q(S',a') - Q(S,A)\right)$$



SARSA

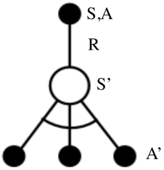
Simplest TD Q update formula is the **S A R S A** update. It is "on-policy". Can you write down the update for Q?

$$Q(S, A) \leftarrow Q(S, A) + \alpha \left(R + \gamma Q(S', A') - Q(S, A)\right)$$



Can you write the **Q-learning** update for Q. It is called off-policy.

$$Q(S, A) \leftarrow Q(S, A) + \alpha \left(R + \gamma \max_{a'} Q(S', a') - Q(S, A)\right)$$

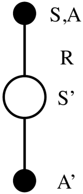


Why is one called on-policy and the other one off-policy?

SARSA

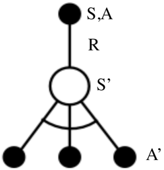
Simplest TD Q update formula is the **S A R S A** update. It is "on-policy". Can you write down the update for Q?

$$Q(S,A) \leftarrow Q(S,A) + \alpha\,(R + \gamma Q(S',A') - Q(S,A))$$



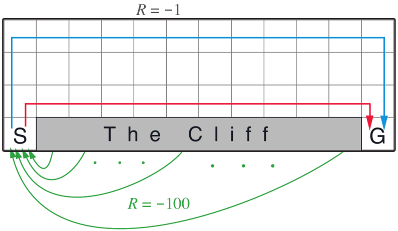
Can you write the **Q-learning** update for Q. It is called off-policy.

$$Q(S,A) \leftarrow Q(S,A) + \alpha\left(R + \gamma \max_{a'} Q(S',a') - Q(S,A)\right)$$



Why is one called on-policy and the other one off-policy? Solution: Sarsa uses the action A' produced by π (on-policy). Q-Learning uses the best local A' for value-backup (off-policy).

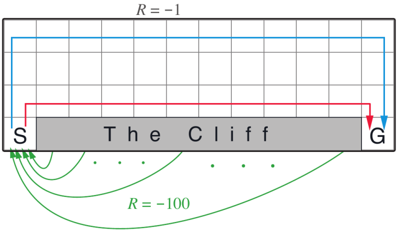
SARSA vs. Q: Cliff Walking Example



Fixed ϵ-greedy exploration policy. No noise in environment.

1. Which path will be learned by Q-Learning?
2. Which path will be learned by SARSA?

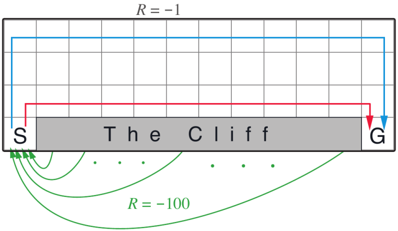
SARSA vs. Q: Cliff Walking Example



Fixed ϵ-greedy exploration policy. No noise in environment.

1. Which path will be learned by Q-Learning?
2. Which path will be learned by SARSA?

Solution: Q: Red, SARSA: Blue



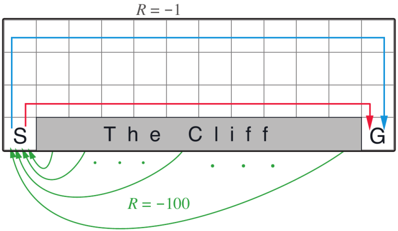
Fixed ϵ-greedy exploration policy. No noise in environment.

1. Which path will be learned by Q-Learning?
2. Which path will be learned by SARSA?

Solution: Q: Red, SARSA: Blue

3. Which solution is better?
   Discuss with your neighbor.

SARSA vs. Q: Cliff Walking Example

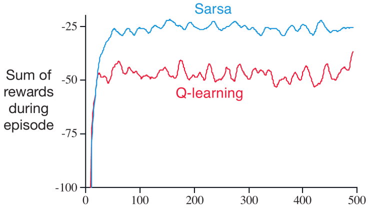


Fixed ϵ-greedy exploration policy. No noise in environment.

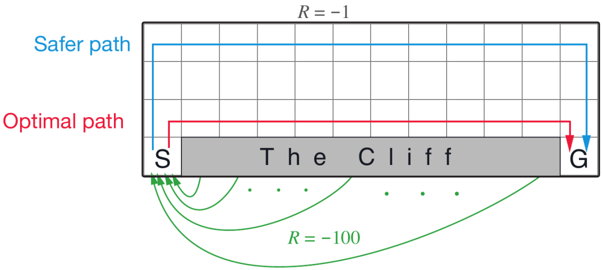
1. Which path will be learned by Q-Learning?
2. Which path will be learned by SARSA?

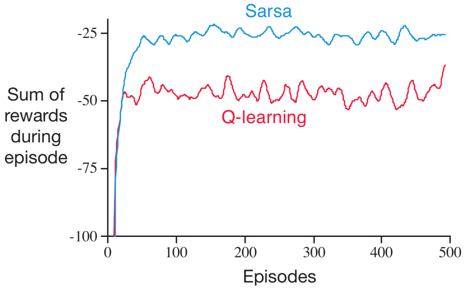
Solution: Q: Red, SARSA: Blue

3. Which solution is better?
   Discuss with your neighbor.
4. Are you sure after seeing these curves?



Cliff Walking Example (Slide from Lecture)

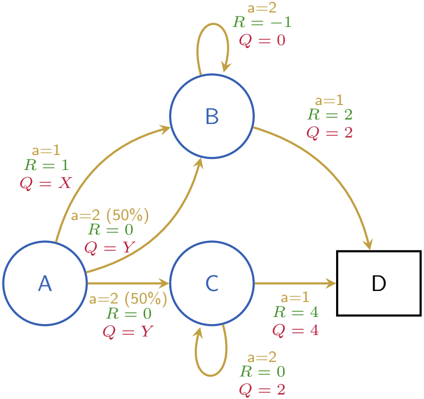




Q-learning learns the optimal policy, but shows lower online performance
(falls occationally off the cliff)

SARSA learns to cope with the ϵ-greedy policy: has to choose a saver but less optimal path.

Q-Update for SARSA and Q

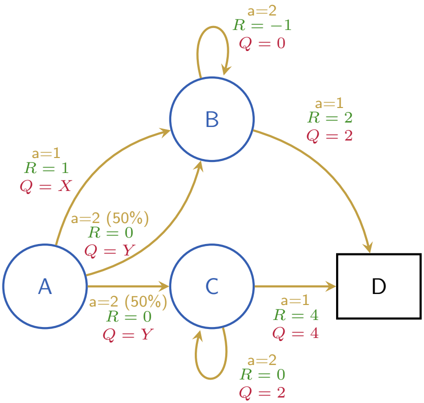


The MDP on the left is unknown to us!

Let's say the Q value for state B,C,D are already estimated. Compute the Q values X,Y (initialized 0) from the following transitions from the ϵ-Greedy policy. $\alpha = 0.5, \gamma = 0.5$

	S	A	R	S'	A'	SARSA: $Q'(S, A)$	Q-learning: $Q'(S, A)$
τ_1	A	2	0	C	2		
τ_2	A	1	1	B	2		
τ_3	A	1	1	B	1		

Q-Update for SARSA and Q



The MDP on the left is unknown to us!

Let's say the Q value for state B,C,D are already estimated. Compute the Q values X,Y (initialized 0) from the following transitions from the ϵ-Greedy policy.
$\alpha = 0.5, \gamma = 0.5$

	S	A	R	S'	A'	SARSA: $Q'(S, A)$	Q-learning: $Q'(S, A)$
τ_1	A	2	0	C	2	0.5 = 0.5(0+0.5*2)	1 = 0.5(0+0.5*4)
τ_2	A	1	1	B	2	0.25 = 0.5*(1 + 0.5*(-1))	1 = 0.5*(1 + 0.5*2)
τ_3	A	1	1	B	1	1.125 = 0.25 + 0.5*(1+0.5*2-0.25)	1.5 = 1 + 0.5*(1+0.5*2 - 1)

