Model-Free Control [2]

DOROZHKO Anton

Novosibirsk State University

May 20, 2020

Novosibirsk State University
*THE REAL SCIENCE

[2] David Silver's Lecture 5

Outline

1. Inroduction
2. On-policy Q-learning
3. Off-policy Q-learning

Markov Decision Process

Definition

A Markov Decision Process is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{A}$ is a finite set of actions
- $\mathcal{P}$ is a transition probability matix

$$\mathcal{P}^a_{ss'} = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a]$$

- $\mathcal{R}$ is a reward function :

$$\mathcal{R}^a_s = \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$$

- γ is a discount factor, $\gamma \in [0, 1]$

THE REAL SCIENCE

Model-free RL

Previous lecture

- Model-free prediction
- Evaluate value function in unknown MDP

This lecture

- Model-free control
- Optimize value function in unknown MDP

For most problems, either:

- MDP is unknown, but experience can be sampled
- MDP is known, but is too big to use, except by samples (Go)

Novosibirsk State University
*THE REAL SCIENCE

On and Off-Policy learning

On-policy learning

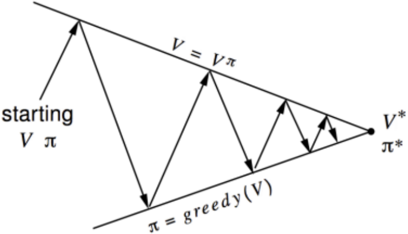
- "Learn on the job"
- Learn about policy π from experience sampled from policy π

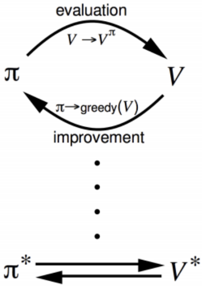
Off-policy learning

- "Look over someone's shoulder"
- Learn about policy π from experience sampled from policy β
- β sometimes called behaviour policy

Novosibirsk State University
*THE REAL SCIENCE

Generalised Policy Iteration

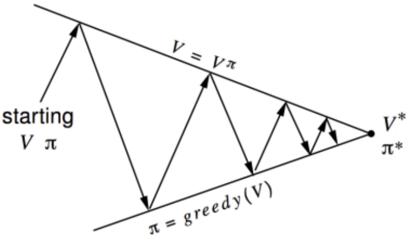




Policy evaluation Estimate V^{π}

Policy improvement Generate $\pi' \geq \pi$

Generalised Policy Iteration with MC



Policy evaluation MC policy evaluation ?
Policy improvement Greedy policy improvement ?

Greedy policy improvement

- Greedy policy improvement over $V(s)$ requires model of MDP

$$\pi'(s) = arg\max_{a \in A} R_s^a + P_{ss'}^a V(s')$$

- Greedy policy improvement over $Q(s, a)$ is model-free

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

Novosibirsk State University
*THE REAL SCIENCE

ϵ-Greedy Exploration

- Simple idea for continual exploration
- All actions are tried with $p > 0$

$$\pi(a|s) = \begin{cases} \epsilon/|A| + 1 - \epsilon & a^* = arg\max_{a \in A} Q(s, a) \\ \epsilon/|A| & otherwise \end{cases}$$

Novosibirsk State University
*THE REAL SCIENCE

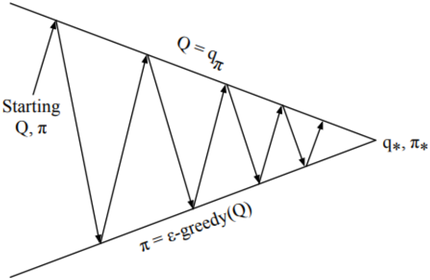
ϵ-Greedy Policy Improvement

Theorem

For any ϵ-greedy policy π, the ϵ-greedy policy π' with respect to q_π is an improvement, $V^{\pi'}(s) \geq V^{\pi}(s)$

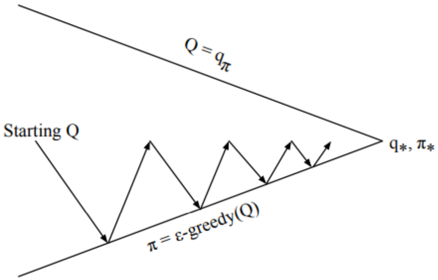
$$
\begin{aligned}
q_\pi(s, \pi'(s)) &= \sum_{a \in A} \pi'(a|s) q_\pi(s, a) \\
&= \frac{\epsilon}{|A|} \sum_{a \in A} q_\pi(s, a) + (1 - \epsilon) \max_{a \in A} q_\pi(s, a) \\
&\geq \frac{\epsilon}{|A|} \sum_{a \in A} q_\pi(s, a) + (1 - \epsilon) \sum \frac{pi(a|s) - \frac{\epsilon}{|A|}}{1 - \epsilon} q_\pi(s, a) \\
&= \sum_{a \in A} \pi(a|s) q_\pi(s, a) = v_\pi(s)
\end{aligned}
$$

MC Policy Iteration



Policy evaluation MC policy evaluation $Q = q_\pi$
Policy improvement ϵ-Greedy policy improvement ?

MC control



Policy evaluation MC policy evaluation $Q \sim q_\pi$
Policy improvement ϵ-Greedy policy improvement ?

GLIE

Definition

Greedy in the Limit with Infinite Exploration (GLIE)

- All state-action pairs are explored infinitely many times

$$\lim_{k \to \infty} N_k(s, a) = \infty$$

- The policy converges on a greedy policy

$$\lim_{k \to \infty} \pi_k(a|s) = Q_k(s, a')$$

Example: $\epsilon_k = \frac{1}{k}$

Novosibirsk State University
*THE REAL SCIENCE

GLIE MC control

- Sample episode using π : $S_1, A_1, R_2, ..., S_T \in \pi$
- For each state S_t and action A_t in the episode

$$N(S_t, A_t) \leftarrow N(S_t, A_t) + 1$$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}(G_t - Q(S_t, A_t))$$

- Improve policy based on new Q

$$\epsilon = \frac{1}{k}$$

$$\pi \leftarrow \epsilon - greedy(Q)$$

Novosibirsk State University
*THE REAL SCIENCE

MC vs TD Control

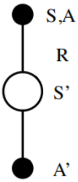
TD advantages over MC

- lower variance
- online
- incomplete sequences

Idea: use TD instead of MC

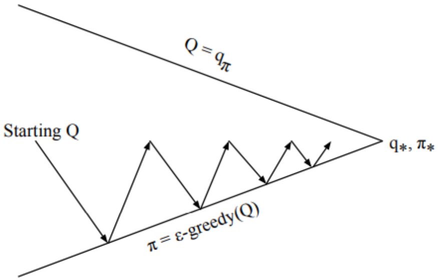
- Apply TD to $Q(S, A)$
- use ϵ-greedy
- update every step

SARSA



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

SARSA control



Policy evaluation SARSA $Q \sim q_\pi$
Policy improvement ϵ-Greedy policy improvement

On-policy SARSA

Initialize $Q(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\textit{terminal-state}, \cdot) = 0$
Repeat (for each episode):
    Initialize S
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Repeat (for each step of episode):
        Take action A, observe R, S'
        Choose A' from S' using policy derived from Q (e.g., ε-greedy)
        $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma Q(S',A') - Q(S,A)\big]$
        $S \leftarrow S'$; $A \leftarrow A'$;
    until S is terminal

Convergence of SARSA

Theorem

Sarsa converges to the oprimal action-values function $Q(s,a) \to q*(s,a)$, under the following conditions:

- *GLIE sequence of policies $\pi_t(a|s)$*
- *Robbins-Monro step-sizes α_t*

$$\sum_{t=1}^{\infty} \alpha_t = \infty$$

$$\sum_{t=1}^{\infty} \alpha_t^2 < \infty$$

*THE REAL SCIENCE

n-step Sarsa

- n-steps return

$$
\begin{aligned}
n &= 1 \quad (Sarsa) \quad & q_t^{(1)} &= R_{t+1} + \gamma Q(S_{t+1}) \\
n &= 2 & q_t^{(2)} &= R_{t+1} + \gamma Q(S_{t+1}) + \gamma^2 Q(S_{t+2}) \\
&. & &. \\
n &= \infty \quad (MC) & q_t^{(T)} &= R_{t+1} + \gamma Q(S_{t+1}) + \gamma^{T-1} R_T
\end{aligned}
$$

- n-step Q-return

$$q_t^(n) = R_{t+1} + \gamma R_{t+2} + ... + \gamma^{n-1} R_{t+n} + \gamma^n Q(S_{t+n})$$

- n-step SARSA updates

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha(q_t^(n) - Q(S_t, A_t))$$

Off-Policy Learning

- Evaluate $\pi(a|s)$ to compute $V^{\pi}(s)$ or $Q^{pi}(s,a)$
- While taking actions by behaviour policy β

$$S_1, A_1, R_2, ..., S_T \sim \beta$$

Profit ?

- Learn from humans / other agents
- Re-use previous experience from π_1, π_2, ... , π_{t-1}
- Learn **optimal** policy following **exploratory** policy
- Learn **multiple** policy followint **one** policy

Importance Sampling

Estimate the expectation of a different distribution

$$\begin{aligned}\mathbb{E}_{X\sim P}[f(X)] &= \sum P(X)f(X)\\ &= \sum Q(X)\frac{P(X)}{Q(X)}f(X)\\ &= \mathbb{E}_{X\sim Q}[\frac{P(X)}{Q(X)}f(X)]\end{aligned}$$

Importance Sampling for Off-policy MC

- Use returns G_t from β to evaluate π
- Weight G_t according to similarity between policies
- along the whole episode

$$G_t^{\pi/\beta} = \frac{\pi(A_t|S_t)}{\beta(A_t|S_t)} \frac{\pi(A_{t+1}|S_{t+1})}{\beta(A_{t+1}|S_{t+1})} \cdots \frac{\pi(A_T|S_T)}{\beta(A_T|S_T)}$$

- update towards corrected return

$$V(S_t) \leftarrow V(S_t) + \alpha(G_t^{\pi/\beta} - V(S_t))$$

- Problem if π is not dominated by β (β is zero, when *pi* is non-zero)
- Increase Variance

Importance Sampling for Off-policy TD

- use TD targets from β to eval π
- use IS to weight TD target $R + \gamma V(S')$
- only 1 IS correction

$$V(S_t) \leftarrow V(S_t) + \alpha \left(\frac{\pi(A_t|S_t)}{\beta(A_t|S_t)} (R_{t+1} + \gamma V(S_{t+1})) - V(S_t) \right)$$

- lower Var than MC IS
- Policies need to be somewhat similar over only a single step

Q-learning

- off-policy learning of $Q(s, a)$
- No IS is required
- Next action $A_{t+1} \sim \beta(.|S_t)$
- Consider alternative successor $A' \sim \pi(.|S_t)$
- update $Q(S_t, A_t)$ towards alternative

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha(R_{t+1} + \gamma Q(S_{t+1}, A') - Q(S_t, A_t))$$

Off-Policy Control with Q-learning

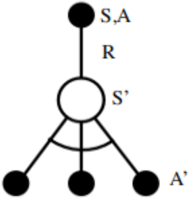
- Improve both behabiour and target
- Target

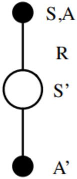
$$\pi(S_{t+1}) = arg \max_{a'} Q(S_{t+1}, a')$$

- Behaviour β ϵ-greedy $Q(s,a))$
- The Q-learning target simplifies:

$$\begin{aligned} & R_{t+1} + \gamma Q(S_{t+1}, A') \\ & = R_{t+1} + \gamma Q(S_{t+1}, arg \max_{a'} Q(S_{t+1}, a') \\ & = R_{t+1} + \max_{a'} \gamma Q(S_{t+1}, a') \end{aligned}$$

Q-learning Control Algorithm

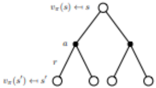
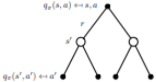
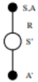
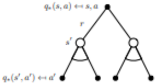
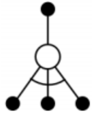




$$Q(S,A) \leftarrow Q(S,A) + \alpha(R + \gamma \max_{a'} Q(S', a') - Q(S,A)$$

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

Relationship between DP and TD

	Full Backup (DP)	*Sample Backup (TD)*
Bellman Expectation Equation for $v_\pi(s)$	 Iterative Policy Evaluation	 TD Learning
Bellman Expectation Equation for $q_\pi(s,a)$	 Q-Policy Iteration	 Sarsa
Bellman Optimality Equation for $q_*(s,a)$	 Q-Value Iteration	 Q-Learning

Relationship between DP and TD

Full Backup (DP)	*Sample Backup (TD)*
Iterative Policy Evaluation $V(s) \leftarrow \mathbb{E}\left[R + \gamma V(S') \mid s\right]$	TD Learning $V(S) \overset{\alpha}{\leftarrow} R + \gamma V(S')$
Q-Policy Iteration $Q(s,a) \leftarrow \mathbb{E}\left[R + \gamma Q(S',A') \mid s,a\right]$	Sarsa $Q(S,A) \overset{\alpha}{\leftarrow} R + \gamma Q(S',A')$
Q-Value Iteration $Q(s,a) \leftarrow \mathbb{E}\left[R + \gamma \max_{a' \in \mathcal{A}} Q(S',a') \mid s,a\right]$	Q-Learning $Q(S,A) \overset{\alpha}{\leftarrow} R + \gamma \max_{a' \in \mathcal{A}} Q(S',a')$

where $x \overset{\alpha}{\leftarrow} y \equiv x \leftarrow x + \alpha(y - x)$