Reinforcement Learning for Business, Economics, and Social Sciences

Unit 4-5: Actor Critic Algorithms

Davud Rostam-Afschar (Uni Mannheim)

Learn acting via policy gradient—
evaluate actions via TD
... or the advantage

Actor Critic

- Q-learning
  - Model-free value-based method
  - No explicit policy representation
- Policy gradient
  - Model-free policy-based method
  - No explicit value function representation
- **Actor Critic**
  - Model-free policy and value based method

Stochastic Gradient Policy
... with a Baseline

Stochastic Gradient Policy Theorem

- Stochastic Gradient Policy Theorem

$$\nabla V_\theta(s_0) \propto \sum_s \mu_\theta(s) \sum_a \nabla \pi_\theta(a|s) Q_\theta(s,a)$$

- Equivalent Stochastic Gradient Policy Theorem with a baseline $b(s)$

$$\nabla V_\theta(s_0) \propto \sum_s \mu_\theta(s) \sum_a \nabla \pi_\theta(a|s) \left[Q_\theta(s,a) - b(s)\right]$$

since $\quad \sum_a \nabla \pi_\theta(a|s) b(s) = b(s) \nabla \sum_a \pi_\theta(a|s) = b(s) \nabla 1 = 0$

Baseline

- Baseline often chosen to be $b(s) \approx V^{\pi}(s)$
- Advantage function: $A(s, a) = Q(s, a) - V^{\pi}(s)$
- Gradient update:

$$\theta \leftarrow \theta + \alpha \gamma^{n} A(s_n, a_n) \nabla \log \pi_{\theta}(a_n | s_n)$$

- Benefit: faster empirical convergence

REINFORCE Algorithm with a baseline

REINFORCE Algorithm with a baseline

REINFORCEwithBaseline(s_0, π_θ)
Initialize π_θ to anything
Initialize V_w to anything
Loop forever (for each episode)

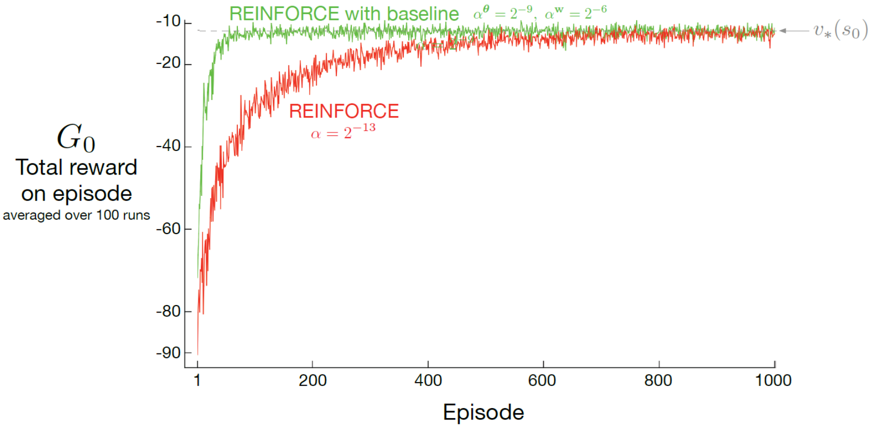
Generate episode $s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_T, a_T, r_T$ with π_θ

Loop for each step of the episode $n = 0, 1, \ldots, T$

$G_n \leftarrow \sum_{t=0}^{T-n} \gamma^t r_{n+t}$
$\delta \leftarrow G_n - V_w(s_n)$
Update value function: $w \leftarrow w + \alpha_w \gamma^n \delta \nabla V_w(s_n)$
Update policy: $\theta \leftarrow \theta + \alpha_\theta \gamma^n \delta \nabla \log \pi_\theta(a_n|s_n)$

Return π_θ

Performance Comparison



Temporal difference update

- Instead of updating $V(s)$ by Monte Carlo sampling

$$\delta \leftarrow G_n - V_w(s_n)$$

- Bootstrap with temporal difference updates

$$\delta \leftarrow r_n + \gamma V_w(s_{n+1}) - V_w(s_n)$$

- Benefit: reduced variance (faster convergence)

Actor Critic Algorithm

ActorCritic(s_0, π_θ)
Initialize π_θ to anything
Initialize Q_w to anything
Loop forever (for each episode)

  Initialize s_0 and set $n \leftarrow 0$

  Loop while s is not terminal (for each time step n)

    Sample $a_n \sim \pi_\theta(a|s_n)$
    Execute a_n, observe s_{n+1}, r_n
    $\delta \leftarrow r_n + \gamma V_w(s_{n+1}) - V_w(s_n)$
    Update value function: $w \leftarrow w + \alpha_w \gamma^n \delta \nabla V_w(s_n)$
    Update policy: $\theta \leftarrow \theta + \alpha_\theta \gamma^n \delta \nabla \log \pi_\theta(a_n|s_n)$
    $n \leftarrow n + 1$

Return π_θ

Advantage update

- Instead of doing temporal difference updates

$$\delta \leftarrow r_n + \gamma V_w(s_{n+1}) - V_w(s_n)$$

- Update with the advantage function

$$A(s_n, a_n) \leftarrow r_n + \gamma \max_{a_{n+1}} Q(s_{n+1}, a_{n+1}) - \sum_a \pi_\theta(a|s_n) Q(s_n, a)$$

$$\theta \leftarrow \theta + \alpha_\theta \gamma^n A(s_n, a_n) \nabla \log \pi_\theta(a_n|s_n)$$

- Benefit: faster convergence

Advantage Actor Critic (A2C)

Advantage Actor Critic (A2C)

A2C(s, π_θ)
Initialize π_θ to anything
Loop forever (for each episode)
Initialize s_0 and set $n \leftarrow 0$

Loop while s is not terminal (for each time step n)

Select a_n
Execute a_n, observe s_{n+1}, r_n
$\delta \leftarrow r_n + \gamma \max_{a'} Q_w(s_{n+1}, a') - Q_w(s_n, a_n)$
$A(s_n, a_n) \leftarrow r_n + \gamma \max_{a'} Q_w(s_{n+1}, a') - \sum_a \pi_\theta(a|s_n) Q_w(s_n, a)$
Update Q: $w \leftarrow w + \alpha_w \gamma^n \delta \nabla_w Q_w(s_n, a_n)$
Update π: $\theta \leftarrow \theta + \alpha_\theta \gamma^n A(s_n, a_n) \nabla \log \pi_\theta(a_n|s_n)$
$n \leftarrow n + 1$

Deterministic Gradient Policy

Continuous Actions

- Consider a deterministic policy $\pi_\theta : s \to a$
- Deterministic Gradient Policy Theorem

$$\nabla V_\theta(s_0) \propto \mathbb{E}_{s\sim\mu_\theta}\left[\nabla_\theta \pi_\theta(s) \nabla_a Q_\theta(s,a)\big|_{a=\pi_\theta(s)}\right]$$

- Proof: see Silver et al. 2014
- Stochastic Gradient Policy Theorem

$$\nabla V_\theta(s_0) \propto \sum_s \mu_\theta(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q_\theta(s,a)$$

Deterministic Policy Gradient (DPG)

Deterministic Policy Gradient (DPG)

DPG(s_0, π_θ)
Initialize π_θ to anything
Loop forever (for each episode)
Initialize s_0 and set $n \leftarrow 0$
    Loop while s is not terminal (for each time step n)
        Select $a_n = \pi_\theta(s_n)$
        Execute a_n, observe s_{n+1}, r_n
        $\delta \leftarrow r_n + \gamma Q_w(s_{n+1}, \pi_\theta(s_{n+1})) - Q_w(s_n, a_n)$
        Update Q: $w \leftarrow w + \alpha_w \gamma^n \delta \nabla_w Q_w(s_n, a_n)$
        Update π: $\theta \leftarrow \theta + \alpha_\theta \gamma^n \nabla_\theta \pi_\theta(s_n) \nabla_a Q_w(s_n, a_n)\big|_{a=\pi_\theta(s_n)}$
        $n \leftarrow n + 1$
Return π_θ

References I

Sigaud, O., and O. Buffet (2013): *Markov decision processes in artificial intelligence*. John Wiley & Sons, Available at https://zodml.org/sites/default/files/Markov_Decision_Processes_and_Artificial_Intelligence.pdf.

Sutton, R. S., and A. G. Barto (2018): "Reinforcement learning: An introduction," *A Bradford Book*, Available at http://incompleteideas.net/book/the-book-2nd.html.

Szepesvári, C. (2022): *Algorithms for reinforcement learning*. Springer nature, Available at https://sites.ualberta.ca/~szepesva/RLBook.html.

Takeaways

Actor Critic algorithms

- Policy gradient methods can be improved with a baseline (value function)
- Actor Critic algorithms use a learned value function as the baseline
- Temporal difference updates reduce variance (faster convergence)
- Deterministic policy gradients can be used for continuous actions