EXPLOITING THE SIGN OF THE ADVANTAGE FUNCTION TO LEARN DETERMINISTIC POLICIES IN CONTINUOUS DOMAINS

{ MATTHIEU.ZIMMER, PAUL.WENG}@SJTU.EDU.CN

Problem

Reinforcement learning is a framework for solving sequential problems where an agent interacts with its environment and adapts its policy based on a reward signal.
We tackle the problem of learning a **deterministic policy** μ_θ with **continuous state and action** spaces in the model-free setting where the exploration strategy is Gaussian: $\pi_{\theta,\sigma}(\cdot|s) = \mathcal{N}\big(\mu_\theta(s), \sigma^2 I\big)$.

Contributions

We provide theoretical explanations and properties of an unorthodox policy update: Continuous Actor-Critic (CAC) [Van Hasselt and Wiering, 2007].
We propose a new **trust region actor-critic algorithm** based on this CAC update, Penalized Neural Fitted Actor-Critic (PeNFAC), which surpasses the state-of-the-art algorithms to learn deterministic policies in several classic control problems.

Policy Gradient

Objective: $J(\theta) = \mathbb{E}\big[\sum_{t=0}^{\infty} \gamma^t r_t\big]$

Iterative policy update: $\theta_{t+1} \leftarrow \theta_t + \alpha \Delta_\theta$

Policy gradient: $\Delta_\theta = \nabla_\theta J(\theta)$

Stochastic Policy Gradient (SPG)
[Sutton *et al.*, 1999]

$$\Delta_{\text{SPG}}(s,\mu_\theta) = A^\pi(s,a)\nabla_\theta \log \pi_\theta(a|s)\Big|_{a\sim\pi_\theta(\cdot|s)}$$

Determinitic Policy Gradient (DPG)
[Prokhorov and Wunsch, 1997; Silver *et al.*, 2014]

$$\Delta_{\text{DPG}}(s,\mu_\theta) = \nabla_a Q^\mu(s,a)\nabla_\theta \mu_\theta(s)\Big|_{a=\mu_\theta(s)}$$

Trust Region

Stochastic: [Kakade and Langford, 2002]

$$\begin{aligned} J(\tilde\pi) =& J(\pi) + \int_{\mathcal{S}} d_\gamma^{\tilde\pi}(s) \int_{\mathcal{A}} \tilde\pi(a|s) A^\pi(s,a)\, da\, ds, \\ \approx& J(\pi) + \int_{\mathcal{S}} d_\gamma^{\pi}(s) \int_{\mathcal{A}} \tilde\pi(a|s) A^\pi(s,a)\, da\, ds. \end{aligned}$$

Deterministic with π as behavior policy:

$$\begin{aligned} J(\tilde\mu) = J(\mu) &+ \int_{\mathcal{S}} d_\gamma^{\pi}(s) \int_{\mathcal{A}} \pi(a|s) A^\mu(s,a)\, da\, ds + \\ &\int_{\mathcal{S}} d_\gamma^{\tilde\mu}(s) A^\pi\big(s,\tilde\mu(s)\big)\, ds, \\ \approx J(\mu) &+ \int_{\mathcal{S}} d_\gamma^{\pi}(s) \int_{\mathcal{A}} \pi(a|s) A^\mu(s,a)\, da\, ds + \\ &\int_{\mathcal{S}} d_\gamma^{\pi}(s) A^\pi\big(s,\tilde\mu(s)\big)\, ds, \end{aligned}$$

Approximation error:

$$\Big| \int_{\mathcal{S}} d^{\tilde\mu}(s) A^\pi\big(s,\tilde\mu(s)\big) - \int_{\mathcal{S}} d^{\pi}(s) A^\pi\big(s,\tilde\mu(s)\big) \Big| \leq \frac{\epsilon L}{1-\gamma}\Big(\big|\big|\tilde\mu(s) - \mu(s)\big|\big|_{2,\infty} + \frac{2m\sigma}{\sqrt{2\pi}} \Big),$$

where $\epsilon = \max_{s,a}|A^\pi(s,a)|$ and L is a Lipschitz constant.

$\rightarrow$ keeps σ small, and $\tilde\mu$ and μ close

Continuous Actor Critic Policy Update

CACLA and CAC reinforce good exploratory actions [Van Hasselt and Wiering, 2007]

$$\Delta_{\text{CAC}}(s,\mu_\theta) = \mathbb{1}_{\delta(s_t,a_t)>0}\, \delta(s_t,a_t)\Big(a_t - \mu_\theta(s_t)\Big)\nabla_\theta \mu_\theta(s_t)\Big|_{a_t\sim\pi_\theta(\cdot|s_t)}$$

Relation with Stochastic Policy Gradient

Rewriting of SPG: $\nabla_\theta J(\pi_{\theta,\sigma}) = \int_{\mathcal{S}} d_\gamma^\pi(s) \int_{\mathcal{A}} \pi_{\theta,\sigma}(a|s) A^\pi(s,a) \nabla_\theta \log \pi_\theta(a|s)\, da\, ds,$

$$= \frac{1}{\sigma^2} \int_{\mathcal{S}} d_\gamma^\pi(s) \int_{\mathcal{A}} \pi_{\theta,\sigma}(a|s) A^\pi(s,a)\big(a - \mu_\theta(s)\big)\nabla_\theta \mu_\theta(s)\, da\, ds.$$

Rewriting of CAC: $\mathbb{1}_{\delta(s_t,a_t)>0}\, \delta(s_t,a_t)\Big(a_t - \mu_\theta(s_t)\Big)\dfrac{\partial \mu_\theta(s_t)}{\partial\theta}\Big|_{a_t\sim\mu_\theta(\cdot|s_t)}$

$$\Longrightarrow \int_{\mathcal{S}} d_\gamma^\pi(s) \int_{\mathcal{A}} \pi_{\theta,\sigma}(a|s)\, \mathbb{1}_{A^\pi(s,a)>0}\, A^\pi(s,a)\big(a - \mu_\theta(s)\big)\nabla_\theta \mu_\theta(s)\, da\, ds.$$

- CAC is only attracted by good actions: moving away from the current deterministic position in continuous spaces will not necessarily lead to an improvement.

Relation with Deterministic Policy Gradient

For a fixed state, when the exploration tends to zero, CAC maintains the sign of the DPG update with scaled magnitudes:

$$\lim_{\sigma\to 0} \Delta_{\text{CAC}}(s,\mu_\theta) = g^+(s,\pi) \circ \Delta_{\text{DPG}}(s,\mu_\theta),$$

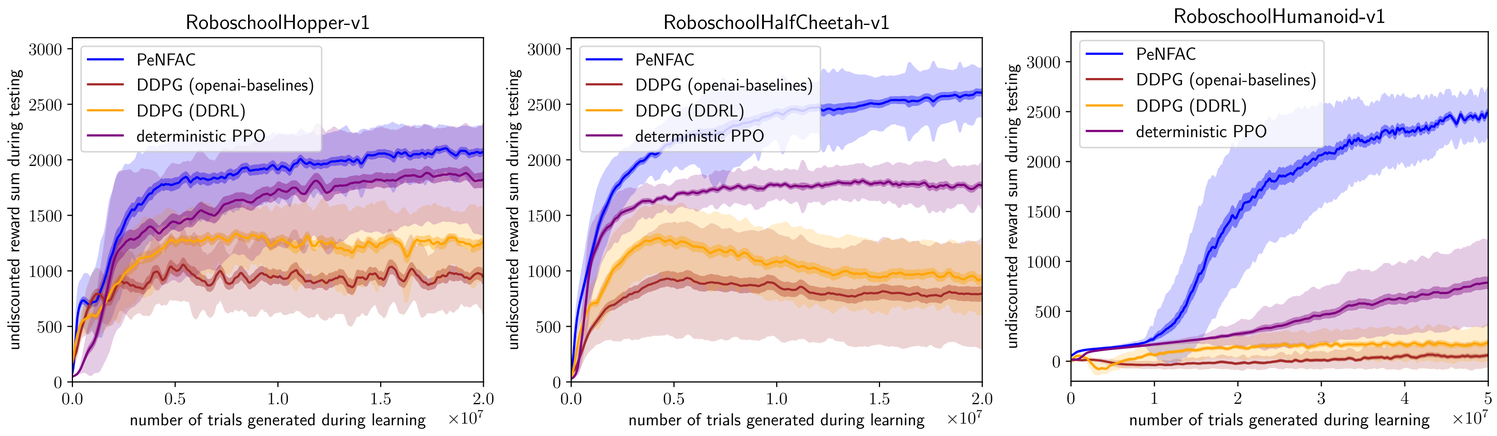
where $g^+(s,\pi)$ is a positive function between $[0;1]^n$ with n as the number of parameters of the deterministic policy and $\circ$ is the Hadamard product.

- DPG needs to learn the action-value function Q whereas the state-value function V is enough for CAC
- DPG makes the assumption that it can access estimate of $\nabla_a Q$
- TD error is a good estimation of the advantage function in nearly deterministic MDPs

Practical Algorithm and Experimental Results

PeNFAC is a batch version of CAC with trust region. The λ-returns are used to estimate the advantage function A and the critic learns the state-value function V.

Actor update: $\sum_{t=0} \Delta_{\text{CAC}}(s_t,\mu_\theta) + \beta\nabla_\theta \big|\big|\mu_{\text{old}}(s_t) - \mu_\theta(s_t)\big|\big|_2^2$



PeNFAC (CAC based) vs DDPG (DPG based) vs PPO (SPG based)

References

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymir Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. *arXiv preprint arXiv:1802.01561*, 2018.

Ehsan Imani, Eric Graves, and Martha White. An off-policy policy gradient theorem using emphatic weightings. In *Advances in Neural Information Processing Systems*, pages 96–106, 2018.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning*, volume 2, pages 267–274, 2002.

Danil V. Prokhorov and Donald C. Wunsch. Adaptive critic designs. *IEEE Transactions on Neural Networks*, 8(5):997–1007, 1997.

David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic Policy Gradient Algorithms. *Proceedings of the 31st International Conference on Machine Learning*, pages 387–395, 2014.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1–2):181–211, 1999.

Hado Van Hasselt and Marco A. Wiering. Reinforcement learning in continuous action spaces. In *Proceedings of the IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning*, pages 272–279, 2007.

Futures Directions

In order to increase **data efficiency**, we want to take into account off-policy updates [Imani *et al.*, 2018] and move to more informed exploration strategy. To improve **computation efficiency**, we want to design distributed implementations [Espeholt *et al.*, 2018].

Source Code

Free open source algorithms and benchmarking problems