ROER: Regularized Optimal Experience Replay

Changling Li, Zhang-Wei Hong, Pulkit Agrawal, Divyansh Garg, Joni Pajarinen

Motivation

Typical prioritized experience replay prioritize out-of-distribution states, likely leading to high value estimation errors.

Method

Introduce f-divergence regularizer

$$\max_{d^{\mathcal{D}}} \mathcal{J}_{D,f}(d^*, d^{\mathcal{D}}) := \mathbb{E}_{(s,a)\sim d^*}[r(s,a)] - \overbrace{\beta}^{\text{Loss temperature}} \overbrace{D_f\left(\underbrace{d^*}_{\text{Optimal on-policy distribution}} \| \underbrace{d^{\mathcal{D}}}_{\text{Buffer off-policy distribution}}\right)}^{\text{f-divergence induced by a convex function } f} \quad D_f(d^* \| d^{\mathcal{D}}) = \mathbb{E}_{(s,a)\sim d^{\mathcal{D}}}[f(\underbrace{w_{*/\mathcal{D}}(s,a)})]$$

$$w_{*/\mathcal{D}} := \frac{d^*(s,a)}{d^{\mathcal{D}}(s,a)}$$

Serves as a penalty when the off-policy deviates too much from on-policy distribution

Experience prioritization as occupancy optimization

Transform the above objective to the following dual problem [1]

$$\tilde{\mathcal{J}}_{D,f}(d^*, d^{\mathcal{D}}) = \min_x \mathbb{E}_{(s,a)\sim d^*}[r(s,a)] + \beta \mathbb{E}_{(s,a)\sim d^{\mathcal{D}}}[\underbrace{f_*(x(s,a))}_{\text{Convex conjugate of } f}] - \beta \mathbb{E}_{(s,a)\sim d^*}[x(s,a)]]$$

Apply change of variable using $Q(s,a) - \gamma V^*(s') = -\beta x(s,a) + r(s,a)$

$$\tilde{\mathcal{J}}_{D,f}(d^*, d^{\mathcal{D}}) = \min_Q \ \beta \cdot \mathbb{E}_{(s,a)\sim d^{\mathcal{D}}}\left[f_*\left((\mathcal{B}^* Q(s,a) - Q(s,a))/\beta\right)\right] + (1-\gamma)\mathbb{E}_{s_0\sim\mu_0, a_0\sim\pi^*(s_0)}\left[Q(s_0, a_0)\right]$$

The solution Q* satisfies $f_*'(\delta_{Q^*}/\beta) = d^*/d^{\mathcal{D}}$

We can shape $d^{\mathcal{D}}$ towards d^* with the weighting formulation

$$\boxed{d^* = f_*'(\delta_{Q^*}/\beta) \cdot d^{\mathcal{D}}}$$

Regularized Optimal Experience Replay

KL Divergence as the Regularizer

- The function of KL divergence has the form $f(x) = x\log(x)$
- Its convex conjugate has the form $f_*(y) = e^y - 1$
- We obtain the objective reminiscent to the loss function of extreme Q-learning [2]

$$\min_Q \ \mathbb{E}_{(s,a)\sim d^{\mathcal{D}}}\left[e^{(\mathcal{B}^* Q(s,a) - Q(s,a))/\beta}\right] - \mathbb{E}_{(s,a,s')\sim d^{\mathcal{D}}}\left[\mathcal{B}^* Q(s,a) - Q(s,a)\right] - 1$$

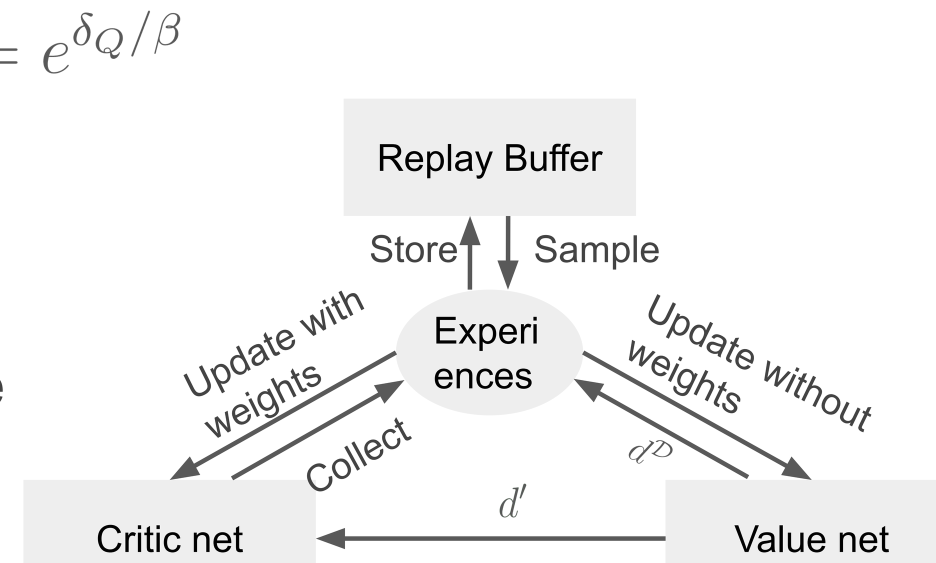
- The occupancy ratio has the form

$$d^*/d^{\mathcal{D}} = f_*'(\delta_Q/\beta) = e^{\delta_Q/\beta}$$

Implementation

- Leverage a separate value network with the regularized objective for plug-in and use as in the right graph
- Solve the optimization in many steps using the following updating rule

$$d' = [\lambda e^{\delta_{Q^*}/\beta} + (1-\lambda)] \cdot d^{\mathcal{D}} \text{ with } \lambda \in (0,1]$$



Conclusion, Limitation & Future Work

Conclusion: We propose a new pipeline of TD error based prioritization scheme and show the relation between the form of priority and the objective function.

Limitation: Additional hyper-parameters and lack of theoretical guarantees

Future work: Adaptive loss temperature and further exploration in offline-to-online fine tuning and offline setting.

Divergence	$f(x)$	$f_*(y)$	$f_*'(y)$
KL	$x\log x$	$e^y - 1$	e^y
Reverse KL	$-\log x$	$-\log(1-y)$	$\frac{1}{1-y}$
Pearson χ^2	$\frac{1}{2}(x-1)^2$	$\frac{1}{2}y^2 + y$	$y+1$
Neyman χ^2	$\frac{(x-1)^2}{2x}$	$-\sqrt{1-2y}+1$	$\frac{1}{\sqrt{1-2y}}$
Total variation	$\frac{1}{2}\lvert x-1\rvert$	y	1
Squared Hellinger	$2(\sqrt{x}-1)^2$	$\frac{2y}{2-y}$	$\frac{4}{(2-y)^2}$

Reference

[1] Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. Advances in neural information processing systems, 32, 2019a
[2] Divyansh Garg, Joey Hejna, Matthieu Geist, and Stefano Ermon. Extreme q-learning: Maxent rl without entropy. arXiv preprint arXiv:2301.02328, 2023
[3] Ilya Kostrikov. JAXRL: Implementations of Reinforcement Learning algorithms in JAX, 10 2021. URL https://github.com/ikostrikov/jaxrl.
[4] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. arXiv preprint arXiv:1511.05952, 2015.
[5] Thibault Lahire, Matthieu Geist, and Emmanuel Rachelson. Large batch experience replay. arXiv preprint arXiv:2110.01528, 2021
[6] Sicen Li, Qinyun Tang, Yiming Pang, Xinmeng Ma, and Gang Wang. Balancing value underestimation and overestimation with realistic actor-critic. arXiv preprint arXiv:2110.09712, 2021.
[7] Haibin Zhou, Zichuan Lin, Junyou Li, Qiang Fu, Wei Yang, and Deheng Ye. Revisiting discrete soft actor-critic. arXiv preprint arXiv:2209.10081, 2022.

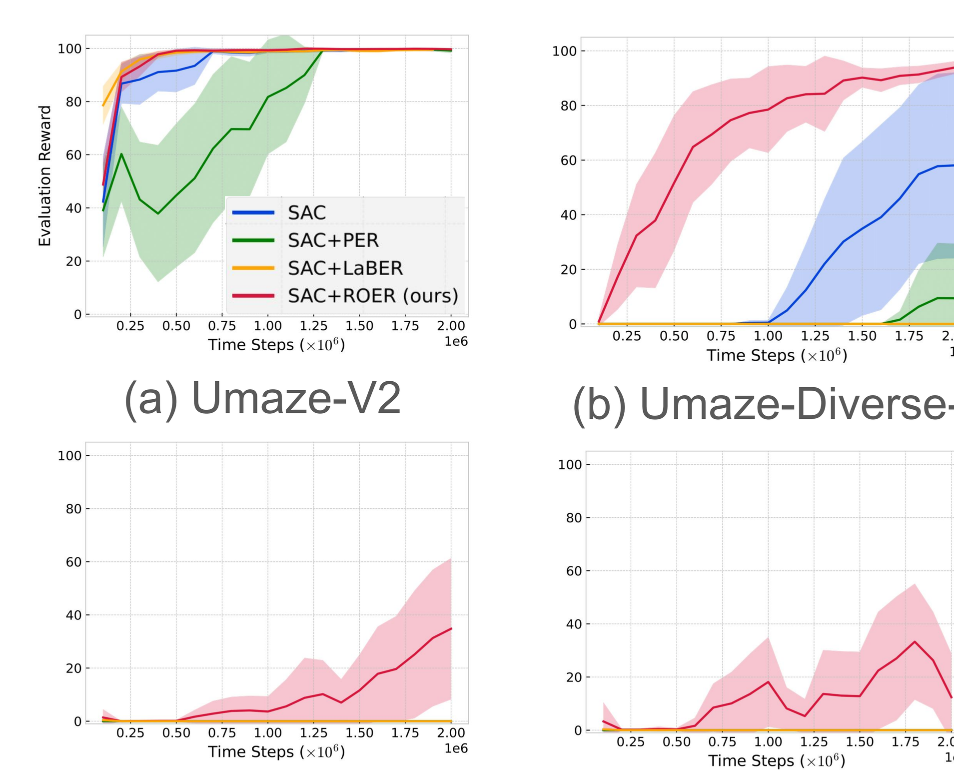
Results

Algorithm: soft-actor critic in JAX [3].
Baselines: uniform experience replay (UER), prioritized experience replay (PER) [4], large batch experience replay (LaBER) [5].

Online: MuJoCo & DM Control

Env	SAC	SAC+PER	SAC+LaBER	SAC+ROER (ours)
Ant-v2	1153.1 ± 335.5	1654.1 ± 342.9	1006.0 ± 546.0	$\mathbf{2275.5} \pm 598.6$
HalfCheetah-v2	9017.4 ± 172.5	9240.4 ± 276.5	7962.8 ± 304.5	$\mathbf{10695.5} \pm 183.4$
Hopper-v2	2813.0 ± 481.2	2937.7 ± 334.3	2330.8 ± 514.3	$\mathbf{3010.2} \pm 299.0$
Humanoid-v2	5026.8 ± 154.1	4993.4 ± 198.0	5000.9 ± 319.5	$\mathbf{5257.0} \pm 153.2$
Walker2d-v2	$\mathbf{4344.3} \pm 177.7$	4003.9 ± 318.7	4033.1 ± 375.7	4328.5 ± 311.4
Fish-swim	247.7 ± 59.6	234.6 ± 63.6	178.3 ± 49.9	$\mathbf{301.9} \pm 54.9$
Hopper-hop	134.4 ± 34.2	$\mathbf{147.2} \pm 31.3$	146.7 ± 29.8	125.7 ± 35.2
Hopper-stand	521.1 ± 120.1	384.7 ± 94.9	475.5 ± 111.0	$\mathbf{798.5} \pm 89.2$
Humanoid-run	130.3 ± 21.7	116.3 ± 18.7	$\mathbf{144.8} \pm 18.1$	137.3 ± 12.3
Humanoid-stand	733.4 ± 53.9	765.0 ± 38.8	$\mathbf{827.8} \pm 40.9$	691.6 ± 57.8
Quadruped-run	761.2 ± 89.4	606.2 ± 114.7	$\mathbf{796.3} \pm 82.6$	772.1 ± 77.7

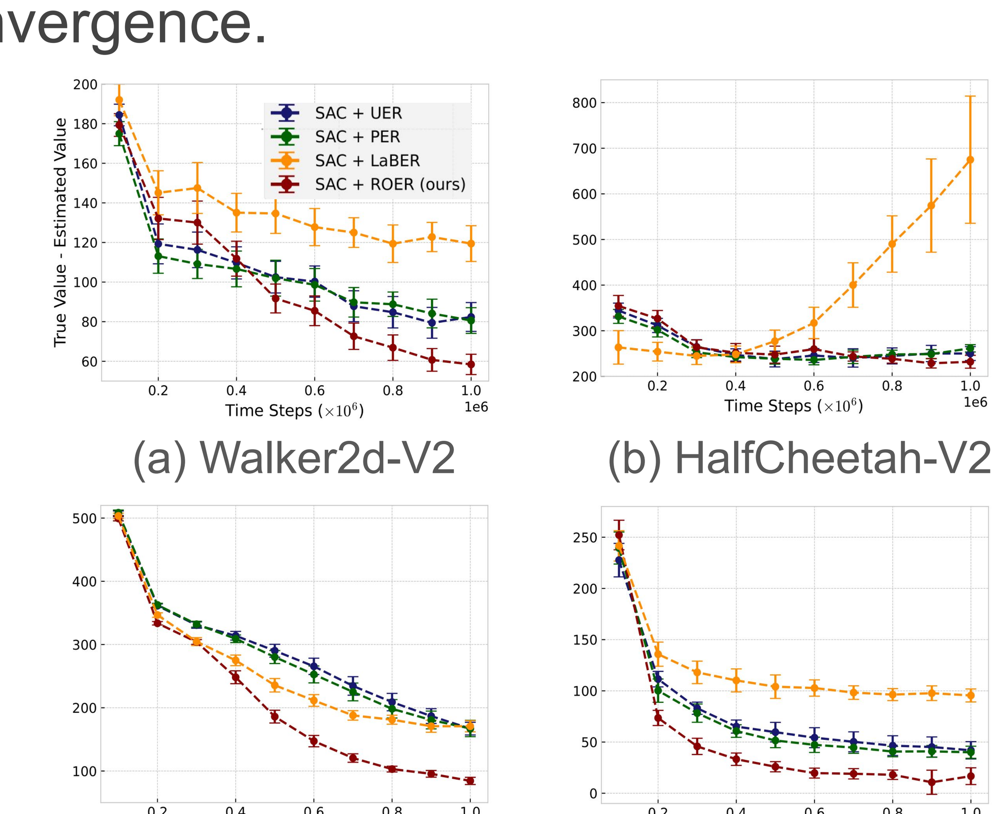
Online with Pretraining: AntMaze D4RL



(a) Umaze-V2 (b) Umaze-Diverse-V2
(c) Medium-Play-V2 (d) Medium-Diverse-V2

Value Estimation Analysis

SAC with double critics tends to underestimate the value [6][7]. ROER shows empirically more accurate value estimation and faster convergence.



(a) Walker2d-V2 (b) HalfCheetah-V2
(c) Humanoid-V2 (d) Hopper-V2

Code Paper