Generalized Data Distribution Iteration

Jiajun Fan, Changnan Xiao

Motivation

Higher Sample Efficiency & Superior Performance

Data Richness

Data diversity is crucial for superior performance.

Exploration-Exploitation Trade-off

Too much exploration hurts the sample efficiency.

Data Distribution Optimization

Superior Exploration & Superior Exploitation

Algorithm 1 Generalized Data Distribution Iteration

Initialize Λ, Θ, $\mathcal{P}_\Lambda^{(0)}$, $\theta^{(0)}$.
for $t = 0, 1, 2, \dots$ **do**
  Sample $\{\mathcal{X}_{\rho_0,\lambda}^{(t)}\}_{\lambda\sim\mathcal{P}_\Lambda^{(t)}}$. {Data Sampling}
  $\theta^{(t+1)} = \mathcal{T}(\theta^{(t)}, \{\mathcal{X}_{\rho_0,\lambda}^{(t)}\}_{\lambda\sim\mathcal{P}_\Lambda^{(t)}})$. {Generalized Policy Iteration}
  $\mathcal{P}_\Lambda^{(t+1)} = \mathcal{E}(\mathcal{P}_\Lambda^{(t)}, \{\mathcal{X}_{\rho_0,\lambda}^{(t)}\}_{\lambda\sim\mathcal{P}_\Lambda^{(t)}})$. {Data Distribution Iteration}
end for

Superior Exploration

Sampling behavior policies π_{θ_λ} from a parameterized policy space that indexed by Λ.

Superior Exploitation

Optimizing a selective distribution $\mathscr{P}_\Lambda$ to maximize some target function $L_\mathscr{E}$.

Superior Guarantee

If we transport more measure on the place with higher value, we have higher expected value.

If $\mathscr{P}_{\Lambda}^{(t+1)}(\lambda) = \mathscr{P}_{\Lambda}^{(t)}(\lambda)\exp(\eta \mathscr{L}_{\mathscr{E}}(\lambda, \theta_{\lambda}^{(t)}))/Z^{(t+1)}$[1], then

$$\mathscr{L}_{\mathscr{T}}(\mathscr{P}_{\Lambda}^{(t+1)}, \theta^{(t+1)}) = \mathsf{E}_{\lambda \sim \mathscr{P}_{\Lambda}^{(t+1)}}[\mathscr{L}_{\mathscr{T}}(\lambda, \theta_{\lambda}^{(t+1)})] \geq \mathsf{E}_{\lambda \sim \mathscr{P}_{\Lambda}^{(t)}}[\mathscr{L}_{\mathscr{T}}(\lambda, \theta_{\lambda}^{(t+1)})] = \mathscr{L}_{\mathscr{T}}(\mathscr{P}_{\Lambda}^{(t)}, \theta^{(t+1)})\,.$$

[1] In paper, we require assumptions 1, 2, 3. This is assumption 2. The other two are assumptions of continuity and co-monotonicity.

Implementation

Soft Entropy Policy Space

$$\theta = (\theta_1, \theta_2), \lambda = (\tau_1, \tau_2, \epsilon),$$

$$\pi_{\theta_\lambda} = \epsilon \cdot \mathrm{Softmax}\left(\frac{A_{\theta_1}}{\tau_1}\right) + (1-\epsilon) \cdot \mathrm{Softmax}\left(\frac{A_{\theta_2}}{\tau_2}\right)$$ [1]

[1] $Q_{\theta_1} = A_{\theta_1} + V_{\theta_1}$, $Q_{\theta_2} = A_{\theta_2} + V_{\theta_2}$. $Q_{\theta_1}, V_{\theta_1}, Q_{\theta_2}, V_{\theta_2}$ are optimized by ReTrace and V-Trace with same/different reward shaping. π_{θ_λ} is optimized by PPO.

Performance

	GDI-H^3	GDI-I^3	Muesli	RAINBOW	LASER	R2D2	NGU	Agent57
Training Scale (Num. Frames)	**2E+8**	**2E+8**	**2E+8**	**2E+8**	**2E+8**	1E+10	3.5E+10	1E+11
Playtime (Day)	**38.5**	**38.5**	**38.5**	**38.5**	**38.5**	1929	6751.5	19290
HWRB	**22**	17	5	4	7	15	8	18
Mean HNS(%)	**9620.33**	7810.1	2538.12	873.54	1740.94	3373.48	3169.07	4762.17
Median HNS(%)	1146.39	832.5	1077.47	230.99	454.91	1342.27	1174.92	**1933.49**
Mean HWRNS(%)	**154.27**	117.98	75.52	28.39	45.39	98.78	76.00	125.92
Median HWRNS(%)	**50.63**	35.78	24.86	4.92	8.08	33.62	21.19	43.62
Mean SABER(%)	71.26	61.66	48.74	28.39	36.78	60.43	50.47	**76.26**
Median SABER(%)	**50.63**	35.78	24.86	4.92	8.08	33.62	21.19	43.62

Hindsight

Highlights & Drawbacks

More General Action Space

In superior guarantee, we do NOT assume any constraint on λ and θ. π_{θ_λ} can be further extended, for instance, a combination with curiosity, planning, option or other techniques.

More Detailed Characteristics

We apply I^k and H^k to character algorithms. k is the dimension of Λ. I and H represents whether all $\theta_\lambda, \lambda \in \Lambda$ are identical. This is still rough.

Thanks!