Model-based Distributional Reinforcement Learning for Risk-sensitive Control

Hao Liang[1,2] and Zhi-Quan Luo[1]

[1]The Chinese University of Hong Kong, Shenzhen [2]Tencent AI Lab

haoliang1@link.cuhk.edu.cn, luozq@cuhk.edu.cn

Main Contributions

- Identify several properties of exponential utility that enables distributional dynamic programming (DDP).
- Propose a optimistic model-based distribution reinforcement learning algorithm, which achieves exponential improvements compared with existing work.
- Correct and tighten the minimax lower bound in [1].
- Provide new technical tool: distributional simulation lemma.

Risk-sensitive Objective

The agent aims to maximize a risk-sensitive objective function

$$V = \frac{1}{\beta}\log(\exp(\beta Z)) = \mathbb{E}[Z] + \frac{\beta}{2}\mathbb{V}[Z] + O(\beta^2)$$

- $\beta > 0$: risk-seeking (favoring high uncertainty in Z)
- $\beta < 0$: risk-averse (favoring low uncertainty in Z)
- $\beta \to 0$: risk-neutral

Key Properties Enabling DDP

- **Additive**: $X \perp Y \Longrightarrow \mathrm{T}(X+Y) = \mathrm{T}(X) + \mathrm{T}(Y)$.
- **Monotonicity preserving (MP)**:
  $\forall\theta \in [0,1], \mathrm{T}(F_1) \geq \mathrm{T}(F_2) \Longrightarrow \mathrm{T}((1-\theta)F_1+\theta G) \geq \mathrm{T}((1-\theta)F_2+\theta G)$.

Distributional Dynamic Programming

The system is identified by a six-tuple $(\mathcal{S}, \mathcal{A}, P, r, H, \mathrm{T})$

Policy Evaluation Ensured by Additivity

$$\eta_h^\pi(s,a)(\cdot) = \Sigma_{s'} P_h(s'|s,a)\eta_{h+1}^\pi(s')(\cdot - r_h(s,a))$$
$$\eta_h^\pi(s) = \eta_h^\pi(s, \pi_h(s))$$

Control Ensured by MP

$$\begin{aligned}
\pi_H(s_H) &= \arg\max_{a_H} r_H(s_H, a_H)\\
\eta_H(s_H) &= \delta(r_h(s_h, \pi_H(s_H)))\\
\eta_h(s_h,a_h)(\cdot) &= \Sigma_{s_{h+1}} P_h(s_{h+1}|s_h,a_h)\eta_{h+1}(s_{h+1})(\cdot - r_h(s_h,a_h))\\
\pi_h(s_h) &= \arg\max_{a_h} \mathrm{T}(\eta_h(s_h,a_h)) = Q_h(s_h,a_h)\\
\eta_h(s_h) &= \eta_h(s_h, \pi_h(s_h))\\
V_h(s_h) &= \mathrm{T}(\eta_h(s_h)) = Q_h(s_h, \pi_h(s_h)).
\end{aligned}$$

Optimistic Transition Distribution Iteration

Algorithm 1 `OTDI`

Inputs: $T = KH$, δ, and r
Initialize and $N_h^1(s,a) := 0$ for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$.
for $k = 1 : K$ **do**
  Receive s_1^k.
  $\eta_H^k(s) = \delta(\max_a r_H(s,a))$.
  for $h = H-1 : 1$ **do**
    $\tilde{P}_h^k = \texttt{Opt\_Tran}(\hat{P}_h^k, \eta_{h+1}^k, N_h^k)$.
    Set $\tilde{T}_h^k = T(\tilde{P}_h^k, r_h)$.
    $\eta_h^k(s,a) = (\tilde{T}_h^k \eta_{h+1}^k)(s,a), \forall s,a$.
    $\pi_h^k(s) = \arg\max_a Q_h^k(s,a) = \mathrm{T}(\eta_h^k(s,a)), \forall s$.
    $\eta_h^k(s) = \eta_h^k(s, \pi_h^k(s))$ and $V_h^k(s) = \mathrm{T}(\eta_h^k(s)), \forall s$.
  end for
  for $h = 1 : H$ **do**
    Take action $a_h^k := \pi_h^k(s_h^k)$ and transit to s_{h+1}^k.
    Update $N_h^{k+1}(s_h^k, a_h^k)$ and $\hat{P}_h^{k+1}(\cdot|s_h^k, a_h^k)$
  end for
end for

Algorithm 2 `Opt_Tran`

Inputs: $\hat{P}_h^k(\cdot|s,a)$, $N_h^k(s,a)$ and η_{h+1}^k.
Compute $V_{h+1}^k(s) = \mathrm{T}(\eta_{h+1}^k), \forall s$.
Sort the states such that $i \leq j$ iff $V_{h+1}^k(i) \leq V_{h+1}^k(j)$.
Output: $O_c((\hat{P}_h^k(\cdot|s,a))$ with $c = \sqrt{\frac{2S}{N_h^k(s,a))}}\log(SAT/\delta)$.

Optimistic Operator O_c

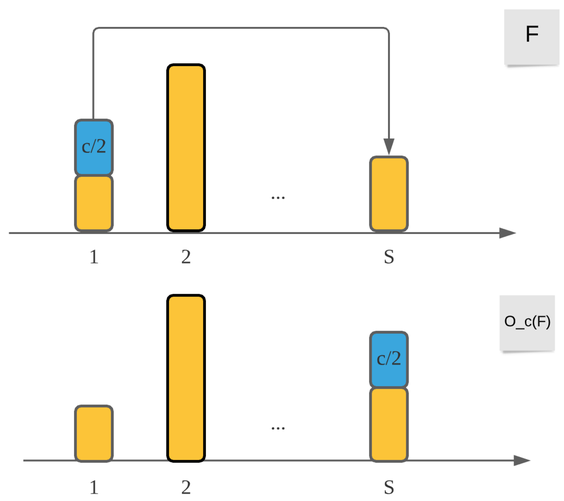

Figure 1:Illustration of O_c on a PMF $F = (F_i)_{i\in[S]}$

Further Properties

- **Lipschitz Continuous**: $|\mathrm{T}(F) - \mathrm{T}(G)| \leq L\|F-G\|_\infty$.
- **Strongly MP**: MP and $\forall 0 \leq \theta \leq \theta' \leq 1, \mathrm{T}(F) \geq \mathrm{T}(G) \Longrightarrow \mathrm{T}((1-\theta)F + \theta G) \geq \mathrm{T}((1-\theta')F + \theta' G)$.
- **Sub-difference**: $\mathrm{T}(F_i) \geq \mathrm{T}(G_i), \forall i \in [n] \Longrightarrow \mathrm{T}(\Sigma\theta_i F_i) - \mathrm{T}(\Sigma\theta_i G_i) \leq C\Sigma\theta_i(\mathrm{T}(F_i) - \mathrm{T}(G_i))$.

Key Parameters

Para.	Upper bound	Tightness
C_M	$\exp(\beta M)$	Tight
L_M	$\frac{\exp(\beta M)-1}{\beta}$	Tight in order

Regret Upper Bound

With high probability, the regret of Algorithm 1 is bounded as

$$\begin{aligned}
\mathrm{Regret}(K) &\leq \tilde{O}(C_{H-1}^{H-2} L_{H-1} H\sqrt{S^2AK})\\
&= \tilde{O}(\exp(\beta H^2)\frac{\exp(\beta H)-1}{\beta} H\sqrt{S^2AK}).
\end{aligned}$$

Regret Lower Bound

For any algorithm π, there exists an non-stationary MDP $\mathcal{M}_\pi$ such that for $K = \Omega(\exp(\beta H/3)HSA)$

$$\mathrm{Regret}(\pi, \mathcal{M}_\pi, K) = \Omega(\frac{\exp(\beta H/6)-1}{\beta H} H\sqrt{SAT}).$$

Distributional Simulation Lemma

Let M and M' be two MDPs satisfy

$$\max_{s,a} ||P_h^M(\cdot|s,a) - P_h^{M'}(\cdot|s,a)||_1 \leq \epsilon_1, \ \forall h \in [H],$$
$$\max_{s,a} |r_h^M(s,a) - r_h^{M'}(s,a)| \leq \epsilon_2, \ \forall h \in [H].$$

Then for every policy π, the two MDPs satisfy

$$\max_{s,a} ||\eta_h^{\pi,M}(s,a) - \eta_h^{\pi,M'}(s,a)||_w \leq (H+1-h)(\epsilon_2 + \frac{H+1-h}{2}\epsilon_1).$$

References

[1] Yingjie Fei, Zhuoran Yang, Yudong Chen, Zhaoran Wang, and Qiaomin Xie.
Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret.
Advances in Neural Information Processing Systems, 33, 2020.