Breaking the Curse of Horizon: Infinite-Horizon Off-Policy Estimation

Qiang Liu†  Lihong Li‡  Ziyang Tang†  Dengyong Zhou‡

† Department of Computer Science, The University of Texas at Austin
‡ Google Brain (KIR)
Off-Policy Reinforcement Learning

**Off-Policy Evaluation**: Evaluate a new policy $\pi$ by only using data from old policy $\pi_0$.

Widely useful when running new RL policies is costly or impossible, due to high cost, risk, or ethics, legal concerns:

- Healthcare
- Robotic & Control
- Advertisement, Recommendation
“Curse of Horizon”

- **Importance Sampling (IS):** Given trajectory \( \tau = \{s_t, a_t\}_{t=1}^T \sim \pi_0, \)

\[
R_\pi = \mathbb{E}_{\tau \sim \pi_0} [w(\tau)R(\tau)], \quad \text{where} \quad w(\tau) = \prod_{t=0}^{T} \frac{\pi(a_t|s_t)}{\pi_0(a_t|s_t)}
\]

- **The Curse of Horizon:**
  - The IS weights \( w(\tau) \) are **product of** \( T \) **terms**; \( T \) is horizon length.
  - Variance can **grow exponentially with** \( T \).
  - **Problematic for infinite horizon problems** (\( T = \infty \)).
Breaking the Curse

**Key: Apply IS on \((s, a)\) pairs, not the whole trajectory \(\tau\):**

\[
R_\pi = \mathbb{E}_{(s,a) \sim d_{\pi_0}} [w(s, a) r(s, a)],
\]

where

\[
w(s, a) = \frac{d_\pi(s, a)}{d_{\pi_0}(s, a)},
\]

where \(d_\pi(s, a)\) is the stationary / average visitation distribution of \((s, a)\) under policy \(\pi\).

**Stationary density ratio** \(w(s, a)\):

- is NOT product of \(T\) terms.
- can be small even for infinite horizon (\(T = \infty\)).
- But is more difficult to estimate.
Main Algorithm

1. Estimate density ratio by a **new minimax objective**:

   \[ \hat{w} = \min_{w \in \mathcal{W}} \max_{f \in \mathcal{F}} \hat{L}(w, f, \mathcal{D}_{\pi_0}) \]

2. Value estimation by IS:

   \[ \hat{R}_\pi = \mathbb{E}_{(s,a) \sim \mathcal{D}_{\pi_0}} [\hat{w}(s, a)r(s, a)] \]

- **Theoretical guarantees** developed for the new minimax objective.
- Can be **kernelized**: Inner max has closed form if \( \mathcal{F} \) is an RKHS.
Empirical Results

Traffic control
(using SUMO simulator\textsuperscript{[5]})

(a) \# of Trajectories ($n$)

(b) Different Behavior Policies

(c) Truncated Length $T$

Naive Average
On Policy (oracle)
WIS Trajectory-wise
WIS Step-wise
Our Method
Thank You!

**Location:** Room 210 & 230 AB; Poster #121  
**Time:** Wed Dec 5th 05:00 – 07:00 PM

References & Acknowledgment


Work supported in part by NSF CRII 1830161 and Google Cloud.