POIS

Policy Optimization via Importance Sampling

Alberto M. Metelli, Matteo Papini, Francesco Faccio and Marcello Restelli
{albertomaria.metelli, matteo.papini, marcello.restelli}@polimi.it, francesco.faccio@mail.polimi.it
POLITECNICO MILANO 1863

Problem and Motivation

- **Reinforcement Learning (RL):** find optimal policy π^*
- **Policy Search**: search over a class of policies π
  - Every policy induces a distribution $p(\cdot|\pi)$ over **trajectories** τ of the Markov Decision Process (MDP)
  - Every trajectory τ has a **return** $R(\tau)$
- **Goal:** find π^* maximizing $J(\pi)$

$$J(\pi) = \mathop{\mathbb{E}}_{\tau \sim p(\cdot|\pi)} [R(\tau)]$$

- **Using data collected with some policy** π:
  - How can I evaluate proposals $\pi' \neq \pi$?
  - How can I trust counterfactual evaluations?
  - How can I best use my data for optimization?

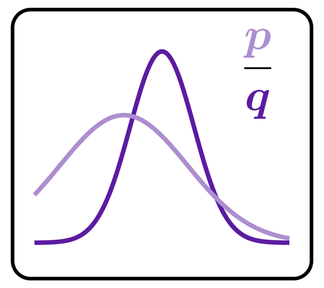
Importance Sampling

How can I evaluate proposals?
With **Importance Sampling (IS)**

- Given a **behavioral** (data-sampling) **distribution** $q(x)$, a **target distribution** $p(x)$, and a function $f(x)$, **estimate**

$$\mu = \mathop{\mathbb{E}}_{x \sim p} [f(x)] \qquad \text{with data from } q \quad x_i \sim q$$

$$\widehat{\mu}_{\text{IS}} = \frac{1}{N} \sum_{i=1}^{N} \underbrace{\frac{p(x_i)}{q(x_i)}}_{w(x_i)} f(x_i)$$



- $w(x) = p(x)/q(x)$ is the **importance weight**
- The estimate is unbiased: $\mathbb{E}_q[\widehat{\mu}_{\text{IS}}] = \mu \ldots$
- ...but the variance can be very high!
- **Rényi divergence**: dissimilarity between p and q:

$$D_2(p\|q) = \log \mathop{\mathbb{E}}_{x \sim q} \left[\left(\frac{p(x)}{q(x)} \right)^2 \right] \qquad \underbrace{d_2(p\|q) = \exp\{D_2(p\|q)\}}_{\text{exponentiated Rényi}}$$

- Variance of the weight depends **exponentially** on the distributional divergence (Cortes et al., 2010)

$$\mathbb{V}\mathrm{ar}[w] = d_2(p\|q) - 1$$

- **Effective Sample Size (ESS)**: number of equivalent samples in plain Monte Carlo estimation ($x_i \sim p$)

$$\text{ESS} = \frac{N}{d_2(p\|q)} \approx \frac{\|w\|_1^2}{\|w\|_2^2} = \widehat{ESS}$$

- Variance of the estimator $\widehat{\mu}_{\text{IS}}$ depends **exponentially** on the distributional divergence as well

$$\mathbb{V}\mathrm{ar}[\widehat{\mu}_{\text{IS}}] \leq \frac{1}{N} \|f\|_\infty^2 d_2(p\|q)$$

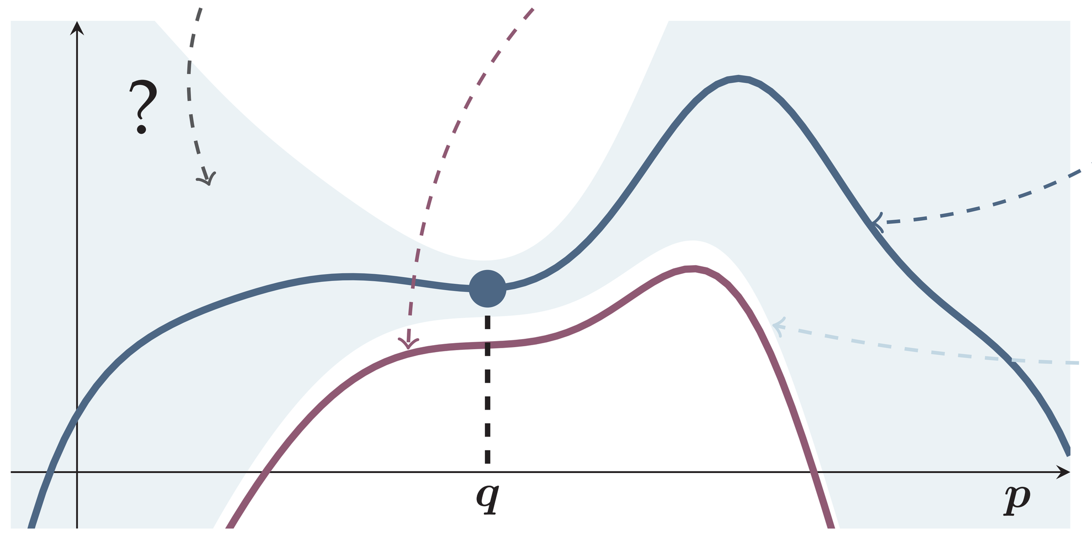
Off-Distribution Learning

How (far) can I trust conterfactual evaluations?

- Evaluate only close solutions: REPS (Peters et al., 2010), TRPO (Schulman et al., 2015)
- **Use a lower bound**: EM (Dayan and Hinton, 1997; Kober et al., 2011), PPO (Schulman et al., 2017), **POIS**

Given a behavioral $q(x)$, a function $f(x)$ and a proposal $p(x)$, with probability at least $1-\delta$:

$$\underbrace{\mathop{\mathbb{E}}_{x \sim p} [f(x)]}_{\text{True function}} \geq \underbrace{\mathcal{L}_\delta^{\text{POIS}}(p/q)}_{\text{Lower Bound}} = \underbrace{\frac{1}{N} \sum_{i=1}^{N} w(x_i) f(x_i)}_{\text{IS Estimator}} - \underbrace{\|f\|_\infty \sqrt{\frac{(1-\delta) d_2(p\|q)}{\delta N}}}_{\text{Variance Bound (Cantelli)}}$$



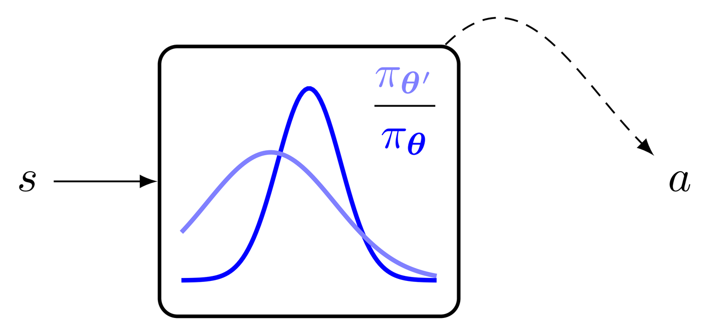
How can I best use my data for optimization?
Given the behavioral q, find p **maximizing** $\mathbb{E}_{x \sim p}[f(x)]$:

1. Collect data with q (expensive in RL)
2. Find p maximizing $\mathcal{L}_\delta^{\text{POIS}}(p/q)$ (offline optimization)
3. Set new behavioral $q \leftarrow p$
4. Repeat until convergence

Action-Based POIS

- Find the **policy** parameters θ^* that maximize $J(\theta')$ (Sutton et al., 2000; Peters and Schaal, 2008)

$$J(\theta) = \mathop{\mathbb{E}}_{\tau \sim p(\cdot|\theta)} [R(\tau)]$$



- Given a **behavioral policy** π_θ we compute a target policy $\pi_{\theta'}$ by optimizing:

$$\mathcal{L}_\lambda^{\text{A-POIS}}(\theta'/\theta) = \frac{1}{N} \sum_{i=1}^{N} \prod_{t=0}^{H-1} \frac{\pi_{\theta'}(a_{\tau_i,t}|s_{\tau_i,t})}{\pi_\theta(a_{\tau_i,t}|s_{\tau_i,t})} R(\tau_i) - \lambda \sqrt{\frac{\widehat{d}_2\left(p(\cdot|\theta')\|p(\cdot|\theta)\right)}{N}}$$

- The term $d_2\left(p(\cdot|\theta')\|p(\cdot|\theta)\right)$ is estimated from samples
- The d_2 grows exponentially with the task horizon H
- λ is a regularization hyperparameter

$$\lambda = \frac{R_{\max}}{1-\gamma} \sqrt{\frac{1-\delta}{\delta}}$$

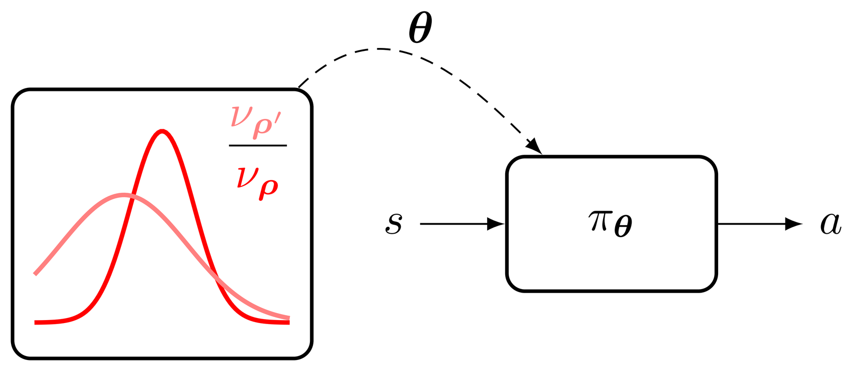
- We consider diagonal Gaussian policies π_θ

$$a \sim \pi_{\mu,\sigma}(\cdot|s) = \mathcal{N}\left(u_\mu(s), \text{diag}(\sigma^2)\right)$$

Parameter-Based POIS

- Find the **hyperpolicy** parameters ρ^* that maximize $J(\rho)$ (Sehnke et al., 2008)

$$J(\rho) = \mathop{\mathbb{E}}_{\theta \sim \nu_\rho} \mathop{\mathbb{E}}_{\tau \sim p(\cdot|\theta)} [R(\tau)]$$



- Given a **behavioral hyperpolicy** ν_ρ we compute a target hyperpolicy $\nu_{\rho'}$ by optimizing:

$$\mathcal{L}_\lambda^{\text{P-POIS}}(\rho'/\rho) = \frac{1}{N} \sum_{i=1}^{N} \frac{\nu_{\rho'}(\theta_i)}{\nu_\rho(\theta_i)} R(\tau_i) - \lambda \sqrt{\frac{d_2\left(\nu_{\rho'}\|\nu_\rho\right)}{N}}$$

- The term $d_2\left(\nu_{\rho'}\|\nu_\rho\right)$ can be computed exactly
- Affected by the parameter space dimension $\dim(\theta)$
- λ is a regularization hyperparameter

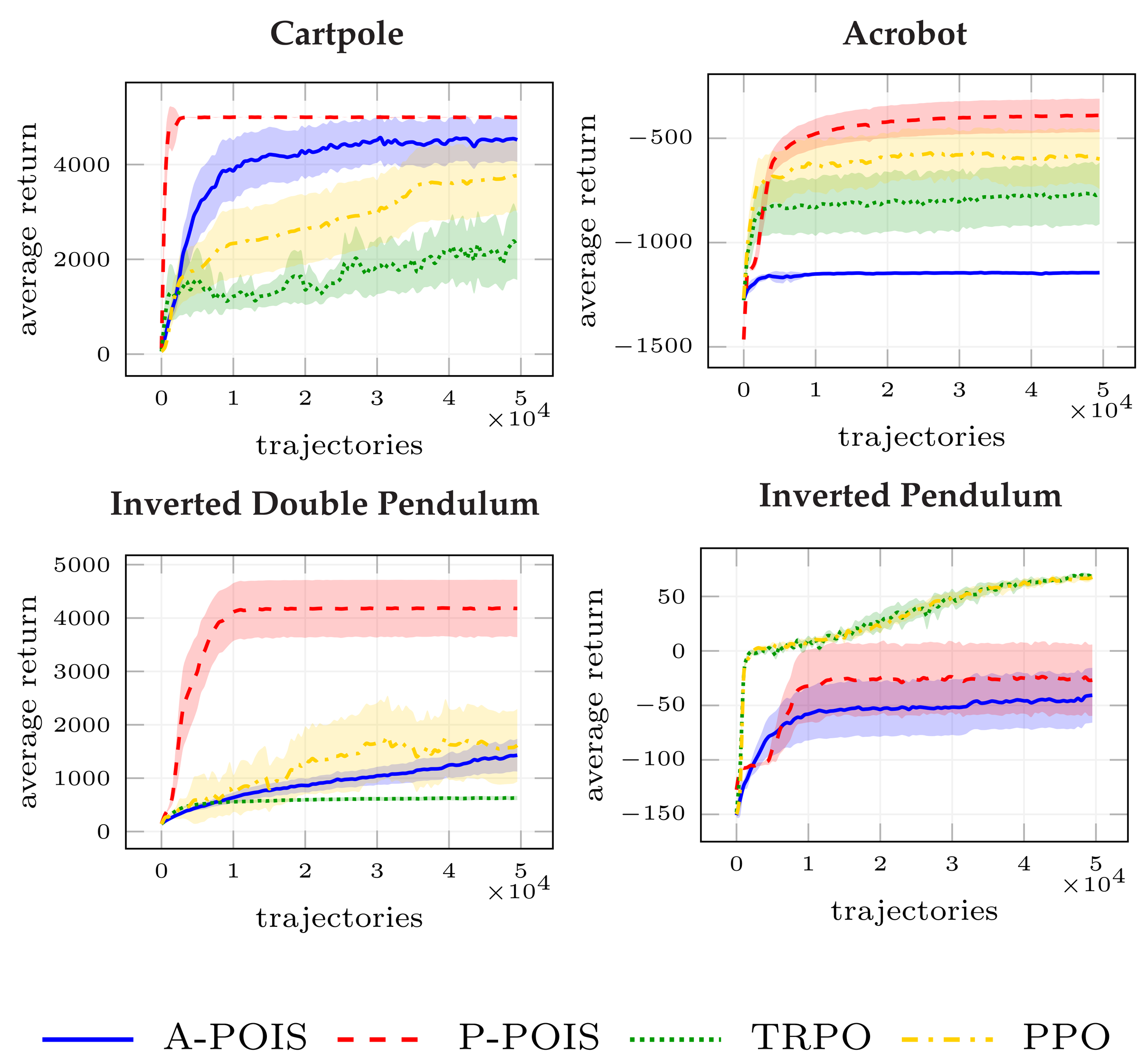
$$\lambda = \frac{R_{\max}}{1-\gamma} \sqrt{\frac{1-\delta}{\delta}}$$

- We consider diagonal Gaussian hyperpolicies ν_ρ

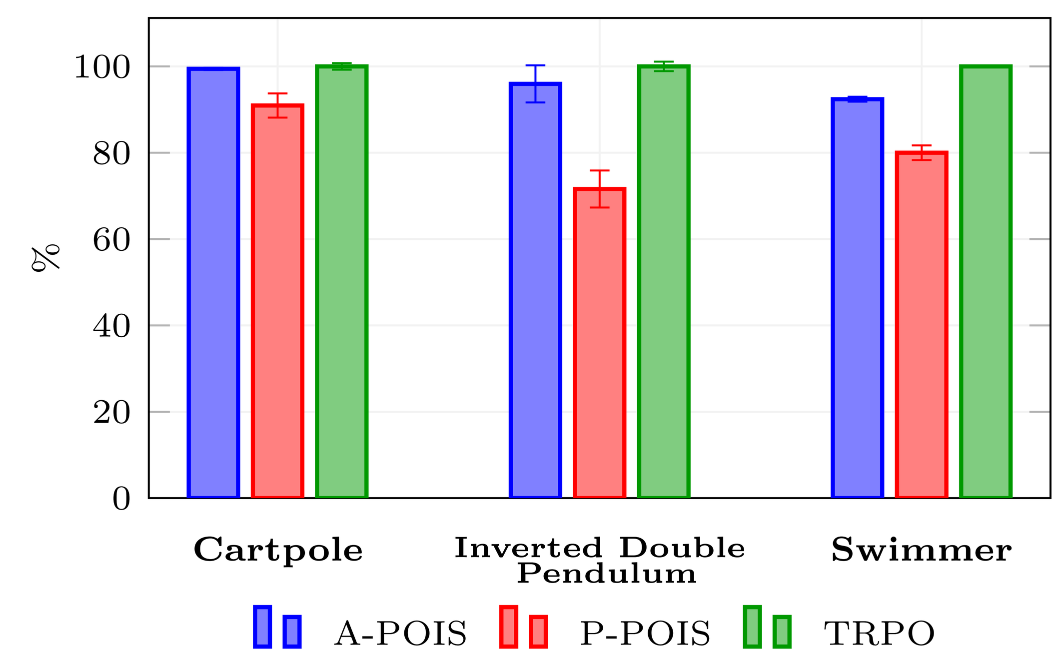
$$\theta \sim \nu_{\mu,\sigma} = \mathcal{N}\left(\mu, \text{diag}(\sigma^2)\right)$$

Experiments

Linear Policies



Deep Policies



Algorithm Details

- **Self-normalized (SN)** importance sampling (Owen, 2013)

$$\widetilde{\mu}_{\text{SN}} = \frac{\sum_{i=1}^{N} w(x_i) f(x_i)}{\sum_{i=1}^{N} w(x_i)} \qquad x_i \sim q$$

- ESS instead of d_2 as penalization
- Gradient optimization of $\mathcal{L}^{\star\text{-POIS}}$ using *line search*
- Natural gradient for P-POIS

References

C. Cortes, Y. Mansour, and M. Mohri. Learning bounds for importance weighting. In *NeurIPS*, 2010.
P. Dayan and G. E. Hinton. Using expectation-maximization for reinforcement learning. *Neural Computation*, 1997.
J. Kober, E. Öztop, and J. Peters. Reinforcement learning to adjust robot movements to new situations. In *IJCAI*, 2011.
A. B. Owen. *Monte Carlo theory, methods and examples*. 2013.
J. Peters and S. Schaal. Reinforcement learning of motor skills with policy gradients. *Neural networks*, 21(4):682–697, 2008.
J. Peters, K. Mülling, and Y. Altun. Relative entropy policy search. In *AAAI*, 2010.
J. Schulman, S. Levine, P. Abbeel, et al. Trust region policy optimization. In *ICML*, 2015.
J. Schulman, F. Wolski, P. Dhariwal, et al. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
F. Sehnke, C. Osendorfer, T. Rückstieß, A. Graves, J. Peters, and J. Schmidhuber. Policy gradients with parameter-based exploration for control. In *International Conference on Artificial Neural Networks*, pages 387–396. Springer, 2008.
R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063, 2000.