Off-Policy Evaluation via Off-Policy Classification

Alex Irpan, Kanishka Rao, Konstantinos Bousmalis, Chris Harris, Julian Ibarz, Sergey Levine

Topic: Imitation - Inverse RL

Presenter: Ning (Angela) Ye

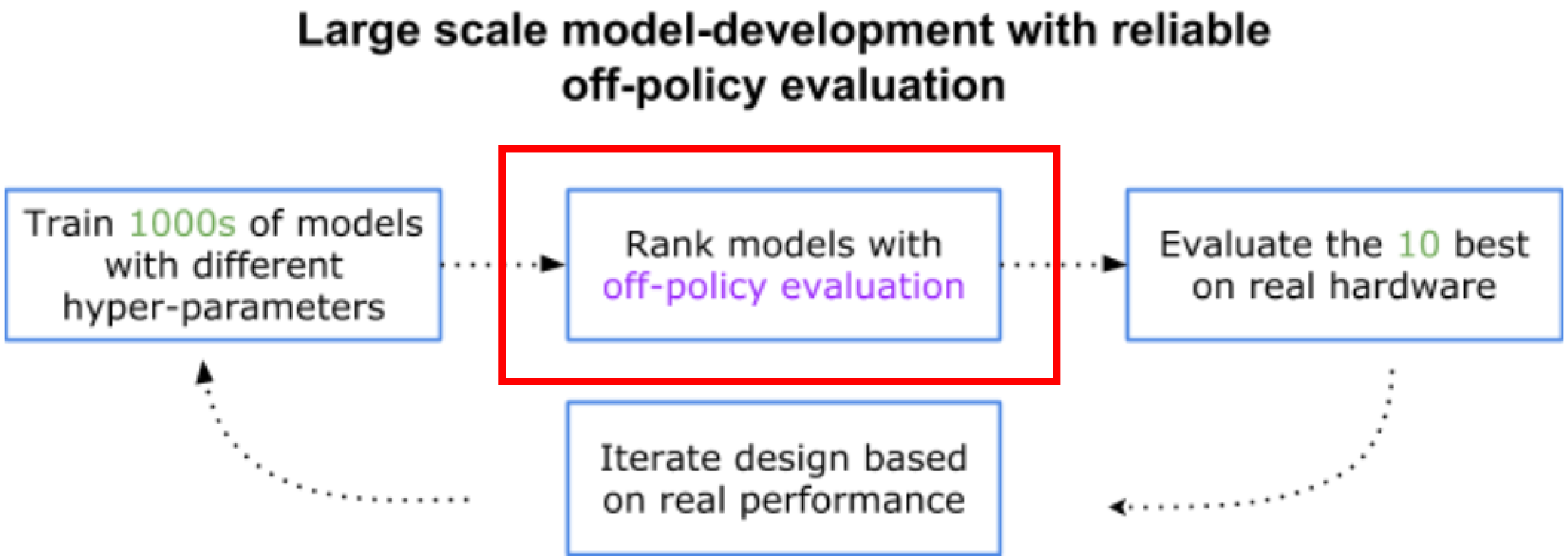
Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

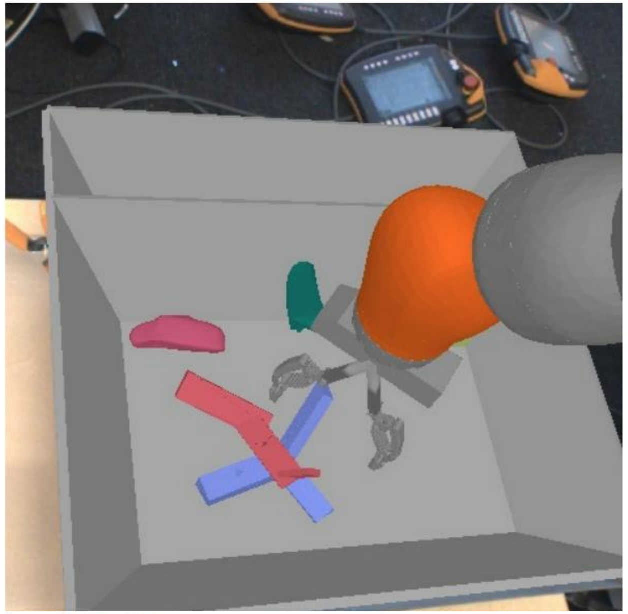
Motivation

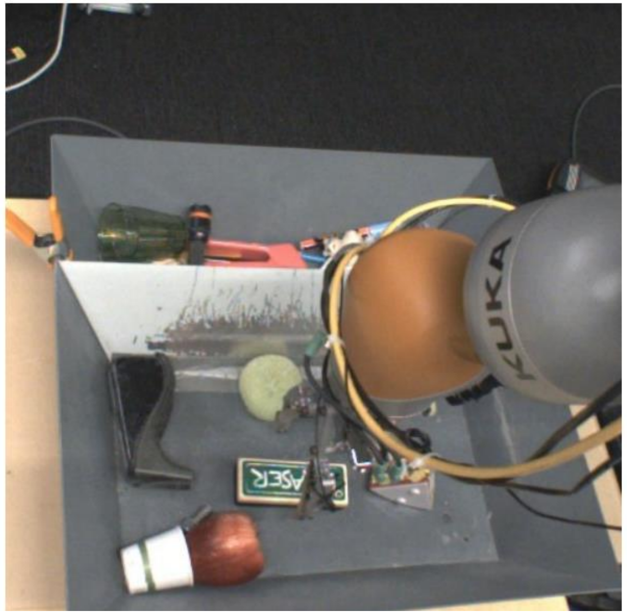


- Typically, performance of deep RL algorithms is evaluated via on-policy interactions
- But comparing models in a real-world environment is costly
- Examines off-policy policy evaluation (OPE) for value-based methods

Motivation (cont.)

- Existing OPE metrics either rely on a model of the environment or importance sampling (IS)
- OPE is most useful in off-policy RL setting, where we expect to use real-world data as “validation set”
  - Hard to use with IS
  - For high-dimensional observations, models of the environment can be difficult to fit





Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

Contributions

- Framed OPE as a positive-unlabeled (PU) classification problem and developed two scores: OPC and SoftOPC
  - Relies on neither IS nor model learning
  - Correlate well with performance (on both simulated and real-world tasks)
- Can be used with complex data to evaluate expected performance of off-policy RL methods
- Proposed metrics outperform a variety of baseline methods including simulation-to-reality transfer scenario

Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

General Background (MDP)

- Focus on finite-horizon Markov decision processes (MDP):

$$(S, A, P, S_0, r, \gamma)$$

- Assume a **binary reward** MDP, which satisfies:
  - $\gamma = 1$
  - Reward is $r_t = 0$ at all intermediate steps
  - Final reward $r_T = \{0,1\}$
- Learn Q-functions $Q(\mathbf{s}, \mathbf{a})$ to evaluate policies

$$\pi(\mathbf{s}) = argmax_{\mathbf{a}} Q(\mathbf{s}, \mathbf{a})$$

General Background (Positive-Unlabeled Learning)

- **Positive-unlabeled** (PU) learning learns binary classification from partially labeled data
  - Sufficient to learn a binary classifier if the positive class prior $p(y = 1)$ is known
- Loss over negatives can be indirectly estimated from $p(y = 1)$

General Background (Positive-Unlabeled Learning)

- Want to evaluate $l(g(x), y)$ over negative examples $(x, y = 0)$

$$p(x) = p(x|y = 1)p(y = 1) + p(x|y = 0)p(y = 0)$$

- Using $\mathbb{E}_X[f(x)] = \int_x p(x)f(x)dx$:

$$\mathbb{E}_X[f(x)] = p(y = 1)\mathbb{E}_{X|Y=1}[f(x)] + p(y = 0)\mathbb{E}_{X|Y=0}[f(x)]$$

- Letting $f(x) = l(g(x), 0)$:

$$p(y = 0)\mathbb{E}_{X|Y=0}[l(g(x), 0)] = \mathbb{E}_{X,Y}[l(g(x), 0)] - p(y = 1)\mathbb{E}_{X|Y=1}[l(g(x), 0)]$$

General Background (Definitions)

- In a binary reward MDP, $(\mathbf{s}_t, \mathbf{a}_t)$ is **feasible** if an optimal π^* has non-zero probability of achieving success after taking $\mathbf{a}_t$ in $\mathbf{s}_t$
- $(\mathbf{s}_t, \mathbf{a}_t)$ is **catastrophic** if even an optimal π^* has zero probability of succeeding after $\mathbf{a}_t$ is taken
- Therefore, return of a trajectory τ is 1 only if all $(\mathbf{s}_t, \mathbf{a}_t)$ in τ are feasible

Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

OPE Method (Theorem)

- Theorem: $R(\pi) \geq 1 - T(\boxed{\epsilon} + \boxed{c})$
  - $\boxed{\epsilon = \frac{1}{T}\sum_{i=1}^{T} \epsilon_t}$ being average error over all $(\mathbf{s}_t, \mathbf{a}_t)$, with
$$\epsilon_t = \mathbb{E}_{\rho_{t,\pi}^{+}}\left[\sum_{\mathbf{a}\in\mathcal{A}\_(\mathbf{s}_t)} \pi(\mathbf{a}|\mathbf{s}_t)\right]$$
  - $\mathcal{A}\_(\mathbf{s})$: set of catastrophic actions at state $\mathbf{s}$
  - $\rho_{t,\pi}^{+}$: state distribution at time t, given that π was followed, and all its previous actions were feasible, and $\mathbf{s}_t$ is feasible
  - $\boxed{c(\mathbf{s}_t, \mathbf{a}_t)}$: probability that stochastic dynamics bring a feasible $(\mathbf{s}_t, \mathbf{a}_t)$ to a catastrophic $\mathbf{s}_{t+1}$, with $c = \max_{\mathbf{s},\mathbf{a}} c(\mathbf{s}, \mathbf{a})$

OPE Method (Missing negative labels)

- Estimate ϵ, probability that π takes a catastrophic action – i.e., $(\mathbf{s}, \pi(\mathbf{s}))$ is a false positive

$$\epsilon = \boxed{p(y=0)\mathbb{E}_{X|Y=0}[l(g(x),0)]}$$

- Recall

$$\boxed{p(y=0)\mathbb{E}_{X|Y=0}[l(g(x),0)]} = \mathbb{E}_{X,Y}[l(\boxed{g(x)},0)] - p(y=1)\mathbb{E}_{X|Y=1}[l(g(x),0)]$$

- We obtain

$$\epsilon = \mathbb{E}_{(\mathbf{s},\mathbf{a})}[l(\boxed{Q(\mathbf{s},\mathbf{a})},0)] - p(y=1)\mathbb{E}_{(\mathbf{s},\mathbf{a}),y=1}[l(Q(\mathbf{s},\mathbf{a}),0)]$$

OPE Method (Off-policy classification)

- **Off-policy classification** (OPC) **score**: negative loss when l is 0-1 loss

$$l(Q(\mathbf{s},\mathbf{a}),Y) = \frac{1}{2} + \left(\frac{1}{2} - Y\right)\operatorname{sign}(Q(\mathbf{s},\mathbf{a}) - \mathrm{b})$$

- **SoftOPC**: negative loss when l is a soft loss function

$$l(Q(\mathbf{s},\mathbf{a}),Y) = (1-2Y)Q(\mathbf{s},\mathbf{a})$$

$$\mathrm{OPC}(Q) = p(y=1)\mathbb{E}_{(\mathbf{s},\mathbf{a}),y=1}\left[1_{Q(\mathbf{s},\mathbf{a})>b}\right] - \mathbb{E}_{(\mathbf{s},\mathbf{a})}\left[1_{Q(\mathbf{s},\mathbf{a})>b}\right]$$

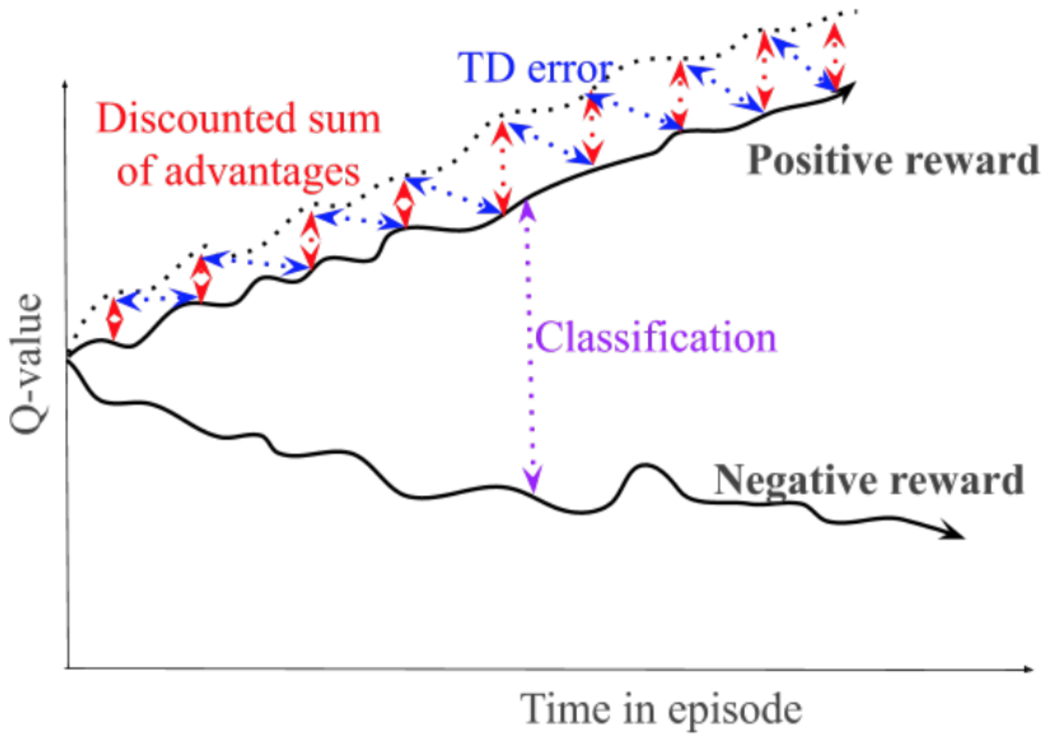
$$\mathrm{SoftOPC}(Q) = p(y=1)\mathbb{E}_{(\mathbf{s},\mathbf{a}),y=1}[Q(\mathbf{s},\mathbf{a})] - \mathbb{E}_{(\mathbf{s},\mathbf{a})}[Q(\mathbf{s},\mathbf{a})]$$

OPE Method (Evaluating OPE metrics)

- Standard method: report MSE to the true episode return
  - Our metrics do not estimate episode return directly
- Instead, train many Q-functions with different learning algorithms
  - Evaluate true return of the equivalent argmax policy for each Q-function
  - Compare correlation of the metric to true return
  - Coefficient of determination of line of best fit R^2, and Spearman rank correlation ξ

Baseline Metrics

- Temporal-difference (TD) error
  - Standard Q-learning training loss
- Discounted sum of advantages $\sum_t \gamma^t A^\pi$
  - Relates $V^{\pi_b}(\mathbf{s}) - V^{\pi}(\mathbf{s})$ to the sum of advantages over data from π_b
- Monte Carlo corrected (MCC) error
  - Arrange discounted sum of advantages into a squared error



Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

Experimental Results (Simple Environments)

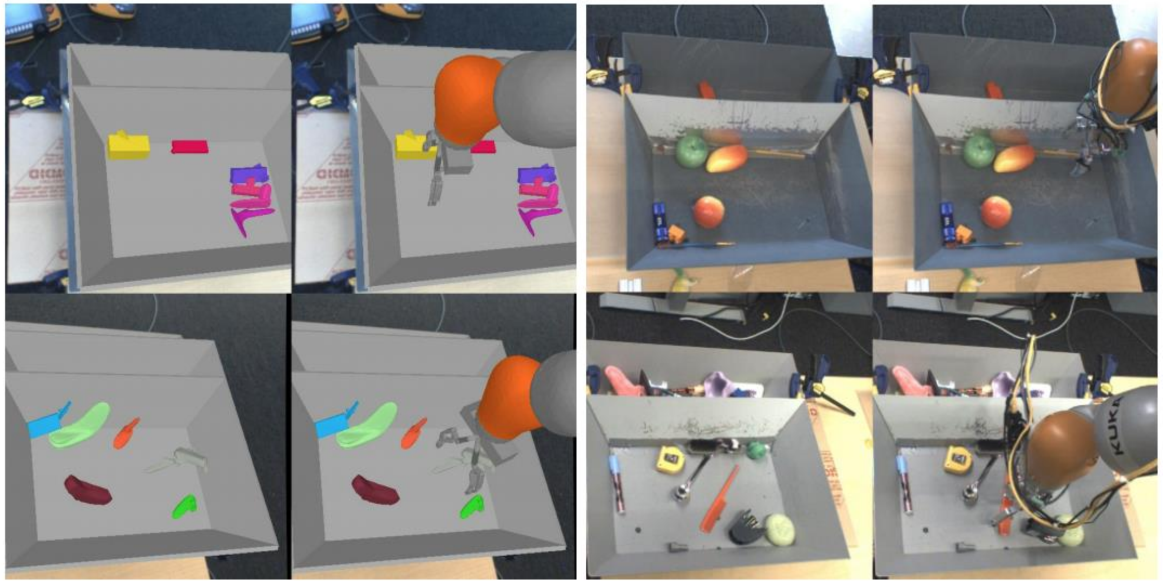
- Performance against stochastic dynamics

	Stochastic Tree 1-Success Leaf						Pong Sticky Actions			
	$\epsilon = 0.4$		$\epsilon = 0.6$		$\epsilon = 0.8$		Sticky 10%		Sticky 25%	
	R^2	ξ	R^2	ξ	R^2	ξ	R^2	ξ	R^2	ξ
TD Err	0.01	-0.07	0.00	-0.05	0.00	-0.05	0.05	-0.16	0.07	-0.15
$\sum \gamma^t A^\pi$	0.00	0.01	0.01	-0.07	0.00	-0.02	0.04	-0.29	0.01	-0.22
MCC Err	0.07	-0.27	0.01	-0.06	0.01	-0.11	0.02	-0.32	0.00	-0.18
OPC (Ours)	0.13	0.38	0.01	0.08	0.03	0.19	**0.48**	**0.73**	**0.33**	**0.66**
SoftOPC (Ours)	**0.14**	**0.39**	**0.03**	**0.18**	**0.04**	**0.20**	0.33	0.67	0.16	0.58

Experimental Results (Vision-Based Robotic Grasping)

	Tree (1 Succ)		Pong		Sim Train		Sim Test		Real-World	
	R^2	ξ	R^2	ξ	R^2	ξ	R^2	ξ	R^2	ξ
TD Err	0.02	-0.15	0.05	-0.18	0.02	-0.37	0.10	-0.51	0.17	0.48
$\sum \gamma^t A^\pi$	0.00	0.00	0.09	-0.32	**0.74**	0.81	**0.74**	**0.78**	0.12	0.50
MCC Err	0.06	-0.26	0.04	-0.36	0.00	0.33	0.06	-0.44	0.01	-0.15
OPC (Ours)	**0.21**	0.50	**0.50**	0.72	0.49	**0.86**	0.35	0.66	0.81	0.87
SoftOPC (Ours)	0.19	**0.51**	0.36	**0.75**	0.55	0.76	0.48	0.77	**0.91**	**0.94**

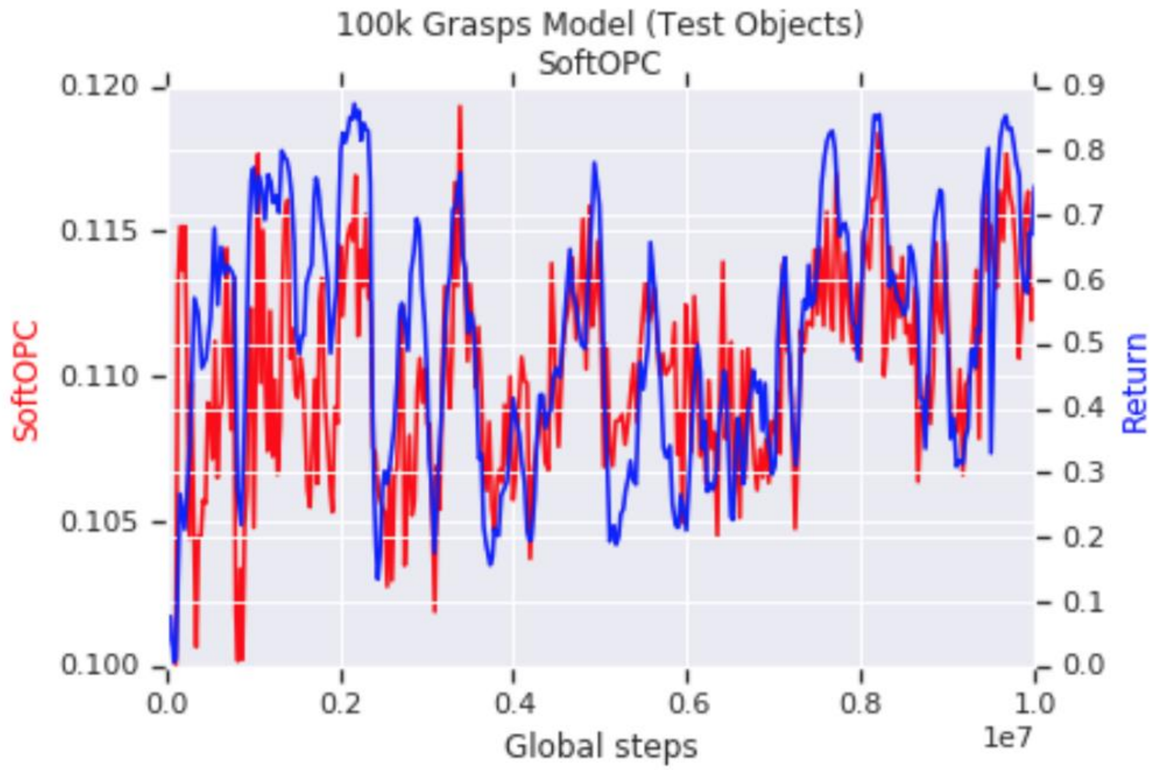
- Performance on simulated and real versions of a vision-based grasping task



(a) Simulated samples (b) Real samples

Discussion of results

- OPC and SoftOPC consistently outperformed baselines
- SoftOPC more reliably ranks policies than baselines for real-world performance
- SoftOPC performs slightly better than OPC



Overview

- Motivation
- Contributions
- Background
- Method
- Results
- Limitations

Limitations

- Key limitation: restricted task domain
  - Assumes an agent either succeeds or fails
  - Difficult to model with complicated tasks with a long time-horizon
- Could not compare to many OPE baselines that use IS and model learning techniques
- High correlation with real-world robotic grasping task, but comparable with sum of discounted advantages in simulation

Contributions (Recap)

- Difficult and expensive to evaluate performance based on real-world environments
  - Many off-policy RL methods are based on value-based methods and do not require any knowledge of the policy that generated the real-world training data
  - These methods are hard to use with IS and model selection
- Treated evaluation as a classification problem and proposed OPC and SoftOPC from negative losses to be used with off-policy Q-learning algorithms
  - Can predict relative performance of different policies in generalization scenarios
- Proposed OPE metrics outperform a variety of baseline methods including simulation-to-reality transfer scenario

Take Home Questions

- What conditions must be met for the MDP to perform OPE via OPC?
- What is a natural choice for the decision function?
- How are the classification scores determined? Which losses are used?
- Which two correlations are used to evaluate the metrics?