UREX: Improving Policy Gradient by Exploring Under-appreciated Rewards

Ofir Nachum Mohammad Norouzi Dale Schuurmans

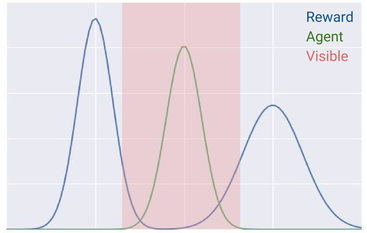
Motivation

Common forms of exploration in reinforcement learning (*e.g.,* epsilon-greedy and entropy regularization) are undirected.

We need smarter, more effective exploration strategies to deal with sparse rewards in high-dimensional action spaces.

Problem

- Optimize a policy $\pi_\theta(\mathbf{a})$ over action sequences to maximize expected reward $\mathbb{E}_{\mathbf{a}\sim\pi_\theta} r(\mathbf{a})$, where $\mathbf{a} \equiv [a_1, a_2, \ldots, a_t]$.
- The reward landscape $r(\mathbf{a})$ is not fully observed.



REINFORCE

Standard policy-based approach to maximize expected reward.

$$\mathcal{O}_{\text{ER}}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{a}\sim\pi_\theta}[\, r(\mathbf{a})\,]$$

- Draw K action sequence samples $\{\mathbf{a}^{(k)}\}_{k=1}^K$ *i.i.d.* from the current policy, *i.e.,* $\mathbf{a}^{(k)} \sim \pi_\theta(\mathbf{a})$ for each $1 \le k \le K$.
- Estimate the gradient

$$\nabla\mathcal{O}_{\text{ER}} = \frac{1}{K}\sum_{k=1}^{K}(r(\mathbf{a}^{(k)}) - b)\,\nabla \log \pi_\theta(\mathbf{a}^{(k)})$$

- Use a baseline b to reduce variance, *e.g.,* sample mean reward.

This fails even on simple problems due to lack of *exploration*.

MENT

Augment the objective with entropy regularization.

$$\mathcal{O}_{\text{MENT}}(\boldsymbol{\theta}, \tau) = \mathbb{E}_{\mathbf{a}\sim\pi_\theta}[\, r(\mathbf{a})/\tau\,] + \mathbb{H}\,(\pi_\theta)$$

- Estimate the gradient using K on-policy samples,

$$\nabla\mathcal{O}_{\text{MENT}} = \frac{1}{K}\sum_{k=1}^{K}(r(\mathbf{a}^{(k)})/\tau \underbrace{-\log \pi_\theta(\mathbf{a}^{(k)})}_{\text{entropy bonus}} - b)\,\nabla \log \pi_\theta(\mathbf{a}^{(k)})$$

MENT does better that REINFORCE, but still fails on problems with a large action space.

We need even better exploration!

Proposal: UREX

The optimal policy that maximizes expected reward is one-hot. Let π^*_τ denote a soft relaxation of the optimal policy:

$$\pi^*_\tau(\mathbf{a}) = \frac{1}{Z}\exp\{r(\mathbf{a})/\tau\} \qquad \left(Z = \sum_{\mathbf{a}'\in\mathcal{A}} \exp\{r(\mathbf{a}')/\tau\}\right)$$

We augment the objective to encourage *mode covering* behavior

$$\mathcal{O}_{\text{UREX}}(\boldsymbol{\theta}, \tau) = \mathbb{E}_{\mathbf{a}\sim\pi_\theta}[\, r(\mathbf{a})/\tau\,] + \mathbb{E}_{\mathbf{a}\sim\pi^*_\tau}[\log \pi_\theta(\mathbf{a})\,]$$

- Draw K *i.i.d.* action sequences $\{\mathbf{a}^{(k)}\}_{k=1}^K$ from $\pi_\theta(\mathbf{a})$.
- Compute self-normalized importance weights

$$\widetilde{w}^{(k)} = \exp\left\{r(\mathbf{a}^{(k)})/\tau - \log \pi_\theta(\mathbf{a}^{(k)})\right\}, \qquad w^{(k)} = \frac{\widetilde{w}^{(k)}}{\sum_{i=1}^K \widetilde{w}^{(i)}}$$

- Estimate the gradient as

$$\nabla\mathcal{O}_{\text{UREX}} = \sum_{k=1}^{K}\Big(\frac{1}{K}(r(\mathbf{a}^{(k)})/\tau - b) + \underbrace{w^{(k)}}_{\text{UREX bonus}}\Big)\nabla \log \pi_\theta(\mathbf{a}^{(k)})$$

Characteristics of UREX

- Rather than undirected exploration, UREX encourages exploration in areas where exponentiated rewards are under estimated by the current policy.
- $\widetilde{w}^{(k)}$ measures the difference between $r(\mathbf{a}^{(k)})/\tau$ and $\log \pi_\theta(\mathbf{a}^{(k)})$, and normalized importance weights find the most **under-appreciated** action sequences among K samples.
- UREX is simple and easy to implement.
- One needs multiple samples to normalize importance weights.

Justification

- Recall KL divergence between distributions $p(\mathbf{a})$ and $q(\mathbf{a})$,

$$\begin{aligned} -D_{\text{KL}}\,(p \parallel q) &= \mathbb{E}_{\mathbf{a}\sim p}[-\log p(\mathbf{a}) + \log q(\mathbf{a})\,] \\ &= \mathbb{H}\,(p) + \mathbb{E}_{\mathbf{a}\sim p}[\log q(\mathbf{a})\,]\,. \end{aligned}$$

- We re-express the entropy regularized objective as a KL,

$$\begin{aligned} -D_{\text{KL}}\,(\pi_\theta \parallel \pi^*_\tau) &= \mathbb{H}\,(\pi_\theta) + \mathbb{E}_{\mathbf{a}\sim\pi_\theta}[\, r(\mathbf{a})/\tau\,] - \log Z \\ &= \mathcal{O}_{\text{MENT}}(\boldsymbol{\theta}, \tau) + C \end{aligned}$$

- $D_{\text{KL}}\,(\pi_\theta \parallel \pi^*_\tau)$ has a *mode seeking* behavior, prone to falling into local minima.
- $D_{\text{KL}}\,(\pi^*_\tau \parallel \pi_\theta)$ has a *mode covering* behavior, but requires sampling from $\pi^*_\tau(\mathbf{a})$.

$$-D_{\text{KL}}\,(\pi^*_\tau \parallel \pi_\theta) = \underbrace{\mathbb{H}\,(\pi^*_\tau)}_{\text{constant}} + \mathbb{E}_{\mathbf{a}\sim\pi^*_\tau}[\log \pi_\theta(\mathbf{a})\,]$$

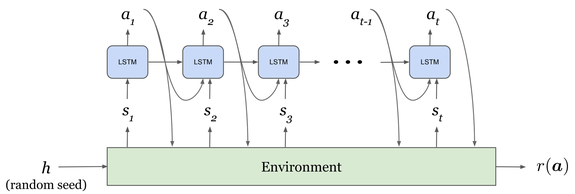
- We use importance sampling to estimate $\nabla D_{\text{KL}}\,(\pi^*_\tau \parallel \pi_\theta)$:

$$w^{(k)} \;\approx\; \pi^*_\tau(\mathbf{a}^{(k)})/\pi_\theta(\mathbf{a}^{(k)})$$

UREX combines

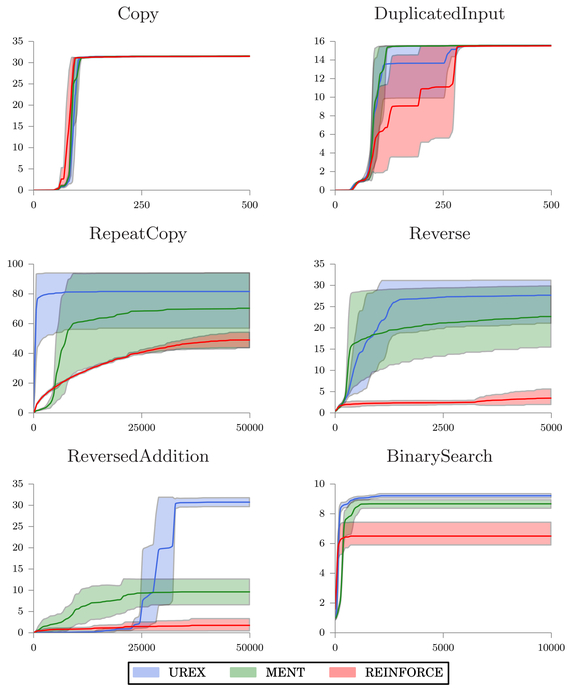
- Mode seeking expected reward objective.
- Mode covering KL between soft optimal policy and current policy.

model



$$\mathbf{a} = [a_1, a_2, \ldots, a_t] \qquad \pi_\theta(\mathbf{a}) = \prod_{i=1}^{t} \pi_\theta(a_i \mid a_{1:i-1}, s_i, h)$$

Experiments



Hyper-parameters

- learning rate $\eta \in \{0.1, 0.01, 0.001\}$
- gradient clipping L2 norm $c \in \{1, 10, 40, 100\}$
- temperature $\tau \in \{0, 0.005, 0.01, 0.1\}$, always $\tau = 0.1$ for UREX

	REINFORCE / MENT				UREX
	$\tau = 0.0$	$\tau = 0.005$	$\tau = 0.01$	$\tau = 0.1$	$\tau = 0.1$
Copy	85.0	88.3	**90.0**	3.3	75.0
DuplicatedInput	68.3	73.3	73.3	0.0	**100.0**
RepeatCopy	0.0	0.0	11.6	0.0	**18.3**
Reverse	0.0	0.0	3.3	10.0	**16.6**
ReversedAddition	0.0	0.0	1.6	0.0	**30.0**
BinarySearch	0.0	0.0	1.6	0.0	**20.0**

Results

- UREX reliably solves reversion and multi-digit addition.
- UREX $\ge$ MENT $\ge$ REINFORCE.
- The RL agents only observe total reward at the end of episode.

	Expected reward		
	REINFORCE	MENT	UREX
Copy	31.2	31.2	31.2
DuplicatedInput	15.4	15.4	15.4
RepeatCopy	48.7	69.2	81.1
Reverse	3.0	21.9	27.2
ReversedAddition	1.4	8.7	30.2
BinarySearch	6.4	8.6	9.1

	Num. of successful attempts out of 5		
	REINFORCE	MENT	UREX
Copy	5	5	5
DuplicatedInput	5	5	5
RepeatCopy	0	3	4
Reverse	0	2	4
ReversedAddition	0	1	5
BinarySearch	0	1	4

Future directions

- Make use of rewards per time step.
- Exploit off-policy samples.
- Exploit expert trajectories.
- Combine with trust region methods.

Softmax Temporal Consistency

Bridging the Gap Between Value and Policy Based RL,
Ofir Nachum, Mohammad Norouzi, Kelvin Xu, Dale Schuurmans,
arXiv:1702.08892, Feb 2017.

- Builds on top of this work to exploit per-step rewards. Proposes a *softmax* temporal consistency between a state-action pair (s, a) and a subsequent state s':

$$Q^*(s,a) = r(s,a) + \gamma\tau \log \sum_{a'} \exp\{Q^*(s',a')/\tau\}$$

References

- See Williams, *et al.* (1991) and Williams (1992) for more on REINFORCE/MENT.
- See Norouzi, *et al.* (2016) for the use of mode covering KL divergence for structured output prediction.
- See Stadie, *et al.* (2015), Schmidhuber (2006), Bellemare, *et al.* (2016) for other exploration strategies.
- Our paper discusses the similarities and differences of UREX with Reward-Weighted Regression (Peters & Schaal, 2007).