Online Learning in Episodic Markovian Decision Processes by Relative Entropy Policy Search

IST AUSTRIA
Institute of Science and Technology

Alexander Zimin
alexander.zimin@ist.ac.at
IST Austria

Gergely Neu
gergely.neu@gmail.com
INRIA Lille – Nord Europe

informatics mathematics

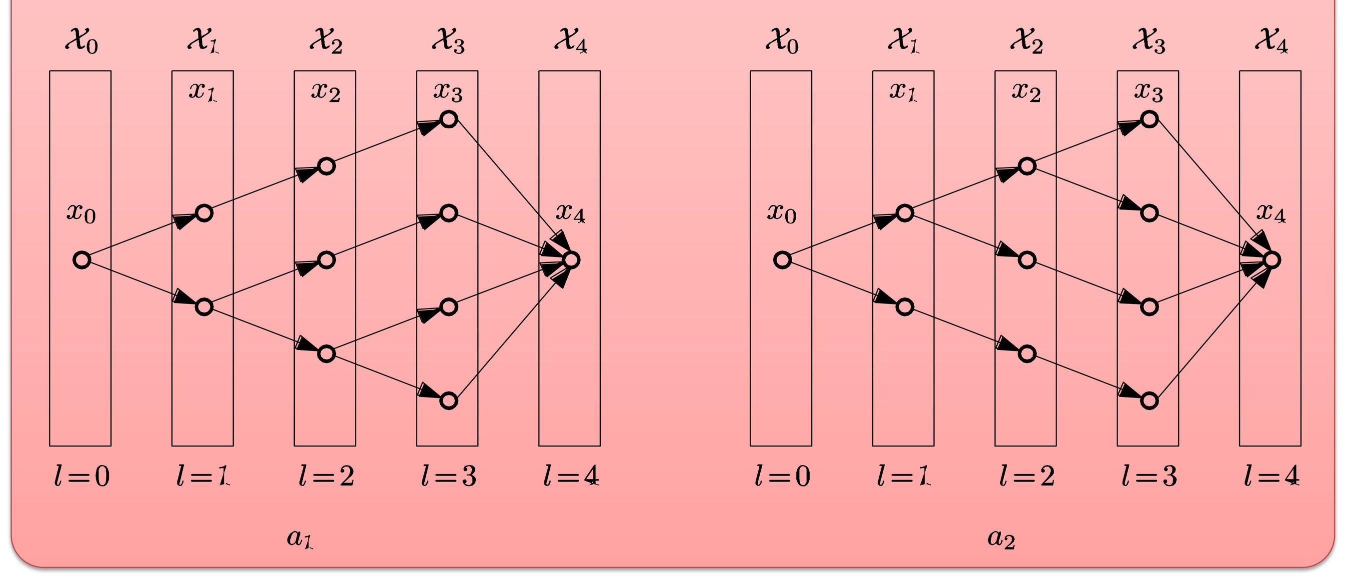
Episodic loop-free MDP

MDP is a tuple $\{X, A, P\}$:

- X: finite known state space
- A: finite known action space
- $P: X \times X \times A \to [0,1]$: transition function

Main assumption:

- Interaction goes in episodes, starts in x_0, ends in x_L
- State space consists of layers, i.e. $X = \cup_{k=0}^{L} X_k$, where $X_k \cap X_j = \emptyset$ for $k \neq j$
- X_0 and X_L are singletons, i.e. $X_0 = \{x_0\}$ and $X_L = \{x_L\}$
- Transitions are possible only between layers, i.e. if $P(x'|x,a) > 0$, then $x' \in X_{k+1}$ and $x \in X_k$ for some k



Online learning in MDP

$\{\ell_t\}_{t=1\dots T}$ – unknown sequence of losses
π_t - policy to follow in episode t

In each of T episodes:

- Start in $x_0(t) = x_0$
  - Observe $x_k(t)$
  - Choose $a_k(t) \sim \pi_t(\cdot\,|x_k(t))$
  - Suffer loss $\ell_t(x_k(t), a_k(t))$
  - State changes to $x_{k+1}(t) \sim P(\cdot\,|x_k(t), a_k(t))$
- Until $x_L(t) = x_L$ is reached
- Observe whole ℓ_t or $\{\ell_t(x_k(t), a_k(t))\}_{k=0\dots L-1}$

Full information

Bandit

$$c_t(\pi) = \mathbb{E}\left\{\sum_{k=0}^{L-1} \ell_t(x_k(t), a_k(t)) \,\middle|\, \pi_t = \pi\right\}$$

Minimize the **regret:**

$$R_T = \max_{\pi} \sum_{t=1}^{T} (c_t(\pi_t) - c_t(\pi))$$

Previous results

Neu et al. (2010):

- Full information: $R_T = O\left(L^2\sqrt{T\log(|A|)}\right)$
- Bandit: $R_T = O\left(\frac{L^2\sqrt{T|A|\log(|A|)}}{\alpha}\right)$, $\alpha > 0$

Reduction to linear optimization

Occupancy measure q^π of a policy π is a family of distributions:

$$q^\pi(x,a) = \mathbb{P}(x'_{k(x)} = x, a'_{k(x)} = a|\pi)$$

layer of x

Δ - set of all such measures

Any q can be computed:

$$\sum_a q^\pi(x,a) = \sum_{x' \in X_{k(x)-1}} \sum_{a'} P(x|x',a')q^\pi(x',a')$$

starting from $q^\pi(x_0, a) = \pi(a|x_0)$

Given q^π we can extract π:

$$\pi(a|x) = {q(x,a)}\Big/{\sum_b q(x,b)}$$

Why are they helpful?

$$c_t(\pi) = \sum_{x,a} q^\pi(x,a)\ell_t(x,a) \stackrel{\text{def}}{=} \langle q^\pi, \ell_t\rangle$$

Instance of online linear optimization:

$$R_T = \max_{q\in\Delta} \mathbb{E}\left[\sum_{t=1}^{T} \langle \boldsymbol{q}_t - q, \ell_t\rangle\right]$$

Estimator of losses

History of interaction:

$$\boldsymbol{u}_t = \{x_k(t), a_k(t), \ell_t(x_k(t), a_k(t))\}_{k=1\dots L-1}$$

The unbiased estimator is

$$\hat{\ell}_t(x,a) = \frac{\ell_t(x,a)}{\boldsymbol{q}_t(x,a)} \mathbb{I}\{(x,a) \in \boldsymbol{u}_t\}$$

Online Relative Entropy Policy Search

Start with uniform policy $\pi_1(a|x) = \frac{1}{|A|}$ and set $q_1 = q^{\pi_1}$

After episode t:

$$\boldsymbol{q}_{t+1}(x,a) = \operatorname*{argmin}_{q\in\Delta}\{\eta\langle q, \ell_t\rangle + D(q|\boldsymbol{q}_t)\}$$

Where D is the unnormalized Kullback–Leibler divergence:

$$D(q|q') = \sum_{x,a} q(x,a)\log\frac{q(x,a)}{q'(x,a)} - \sum_{x,a}(q(x,a) - q'(x,a))$$

And η is a parameter of the algorithm

Implementation

For any $v: X \to R$ and loss $\ell: X \times A \to R$ define

$$\delta(x,a|v,\ell) = -\eta\ell(x,a) - \sum_{x'\in X} v(x')P(x'|x,a) + v(x)$$

Then $\boldsymbol{q}_{t+1}$ can be computed as:

$$\boldsymbol{q}_{t+1}(x,a) = \frac{\boldsymbol{q}_t(x,a)e^{\delta(x,a|\hat{v}_t,\hat{\ell}_t)}}{\boldsymbol{Z}_t(\hat{v}_t, k(x))}$$

Where

$$\hat{v}_t = \operatorname*{argmin}_{v} \sum_{k=0}^{L} \boldsymbol{Z}_t(v,k) \qquad \boldsymbol{Z}_t(v,k) = \sum_{x'\in X_k}\sum_a \boldsymbol{q}_t(x,a)e^{\delta(x,a|v,\hat{\ell}_t)}$$

Results

Full information

$$R_T \le 2L\sqrt{T\log\frac{|X||A|}{L}}$$

Bandit

$$R_T \le 2\sqrt{L|X||A|T\log\frac{|X||A|}{L}}$$