Large-Scale MDPs with KL Control Cost and its Application to Crowdsourcing

Yasin Abbasi-Yadkori, Peter L. Bartlett, Xi Chen, Alan Malek

Berkeley UNIVERSITY OF CALIFORNIA

Motivation

- Markov decision process: modeling sequential decisions
- E.g. queueing network, robot planning
- Dynamic Programming can solve for small state problems
- Applications can have *large* state spaces
- Here: in the KL-cost setting, can efficiently do large state spaces

Notation

An MDP is defined by:

- State space $\mathcal{X} = \{1, \dots, X\}$
- Action space $\mathcal{A}$
- Transition Kernel $K : \mathcal{X} \times \mathcal{A} \to \triangle_{\mathcal{X}}$
- Loss function $\ell : \mathcal{X} \times \mathcal{A} \to [0,1]$

The problem is to:

- Policy $\pi : \mathcal{A} \to \triangle_{\mathcal{A}}$
- Find policy to minimize

$$\boldsymbol{J}_\pi(x) = \mathbb{E}\left[\sum_{t=0}^{\infty} \ell(X_t, \pi) | X_0 = x\right]$$

Aim for optimality *within a restricted family of policies.*

Extending to large state spaces

- Parametric class of policies π_θ for $\theta \in \Theta$ with losses $\boldsymbol{J}_\theta$
- Bellman operator:

$$(\boldsymbol{L}\boldsymbol{J}_\theta)(x) = \min_{a \in \mathcal{A}} \left(\ell(x, a) + \mathbb{E}_{K(x,a)}[\boldsymbol{J}_\theta(x')|x]\right)$$

- Optimal policy has $\boldsymbol{L}\boldsymbol{J}_\theta = \boldsymbol{J}_\theta$.
- Linear Programming formulation:

$$\min_\theta c^\top \boldsymbol{J}_\theta\,, \qquad \text{(low cost)}$$

$$\text{s.t.} \quad \boldsymbol{L}\boldsymbol{J}_\theta \le \boldsymbol{J}_\theta \qquad (\boldsymbol{J}_\theta \text{ is feasible})$$

- Look for efficient relaxations, e.g.

$$\min_\theta c^\top \boldsymbol{J}_\theta + \|\boldsymbol{L}\boldsymbol{J}_\theta - \boldsymbol{J}_\theta\|$$

- Previous ADP techniques used $\boldsymbol{J}_\theta = \Psi\theta$

Linearly Solvable MDPs from [Todorov,

- $\mathcal{A} = \triangle_{\mathcal{X}}$
- Loss: $\ell(x, P(\cdot|x)) = q(x) + D_{KL}(P(\cdot|x)||P_0(\cdot|x))$
  - state loss $q(x)$, base dynamics P_0
  - infinite loss unless $P \ll P_0$
- Terminal state z: $q(z) = 0$ and $P_0(z|z) = 1$
- Total cost of policy P

$$\boldsymbol{J}_P(x) = \mathbb{E}\left[\sum_{t=0}^{\infty} \ell(X_t, P) | X_0 = x\right]$$

- Greedy action is:

$$P_{\boldsymbol{J}}(\cdot|x) = \operatorname*{arg\,min}_{p \in \triangle_{\mathcal{X}}} \boldsymbol{J}_p(x) \propto \frac{1}{Z(x)} P_0(x'|x) e^{-\boldsymbol{J}_P(x')}$$

- Bellman's operator becomes linear in $g(x) = e^{-\boldsymbol{J}(x)}$:

$$e^{\boldsymbol{L}\boldsymbol{J}(x)} = e^{q(x)} \sum_{x'} P_0(x, x') e^{-\boldsymbol{J}(x')}$$

Large state spaces for LMDPs

- Intuition: take $\boldsymbol{J}_\theta = -\log(\Psi\theta)$ so $e^{\boldsymbol{L}\boldsymbol{J}_\theta}$ is linear in θ
- Approximate unconstrained optimization:

$$\min_\theta c^\top \boldsymbol{J}_\theta + H\,\|\boldsymbol{L}\boldsymbol{J}_\theta - \boldsymbol{J}_\theta\|$$

- $\|\boldsymbol{L}\boldsymbol{J}_\theta - \boldsymbol{J}_\theta\|$ not convex in θ, but

$$e^{-max\{\boldsymbol{L}\boldsymbol{J}_\theta, \boldsymbol{J}_\theta\}} \|\boldsymbol{L}\boldsymbol{J}_\theta - \boldsymbol{J}_\theta\| \le \left\|e^{-\boldsymbol{L}\boldsymbol{J}_\theta} - e^{-\boldsymbol{J}_\theta}\right\|$$

- Optimization relaxed to:

$$\min_\theta -c^\top \log(\Psi\theta) + H\left\|e^{-q} P_0 \Psi\theta - \Psi\theta\right\|$$

Our Algorithm for Total Cost

Input: x_1, N, H, step sizes (η_t), v.
Initialize $\theta_1 = \mathbf{0}$.
for $t := 1, 2, \dots, N$ **do**
  Sample trajectory $(x_1, a_1, \dots, z) \sim v$.
  Compute the stochastic subgradient r_t.
  Update $\theta_{t+1} = \Pi_{\mathcal{W}}(\theta_t - \eta_t r_t)$.
end for
$\widehat{\theta}_T = \frac{1}{T}\sum_{t=1}^T \theta_t$.
Return policy $P_{\boldsymbol{J}_{\widehat{\theta}_T}}$

Performance Bound

Theorem 1. *Choose $H \ge e^{\|q\| + \log\|1/\Psi\|}$. Let $\widehat{\theta}$ be an ϵ-optimal solution. Then, for any $\theta \in \Theta$ with $l_\theta = \min(\boldsymbol{J}_\theta, \boldsymbol{L}\boldsymbol{J}_\theta)$,*

$$\boldsymbol{J}_{P_{\boldsymbol{J}_{\widehat{\theta}}}}(x_1) \le \inf_{\boldsymbol{J}_\theta \in \mathcal{J}} \left\{\boldsymbol{J}_{P_{\boldsymbol{J}_\theta}}(x_1) + \mathcal{E}(\boldsymbol{J}_\theta)\right\} + \epsilon + \left\|P_{\boldsymbol{J}_{\widehat{\theta}}} - Q\right\|_1 \max_{T \in \mathcal{T}} \sum_{x \in T} \left|\boldsymbol{J}_{\widehat{\theta}}(x) - \boldsymbol{L}\boldsymbol{J}_{\widehat{\theta}}(x)\right|$$

The penalty function

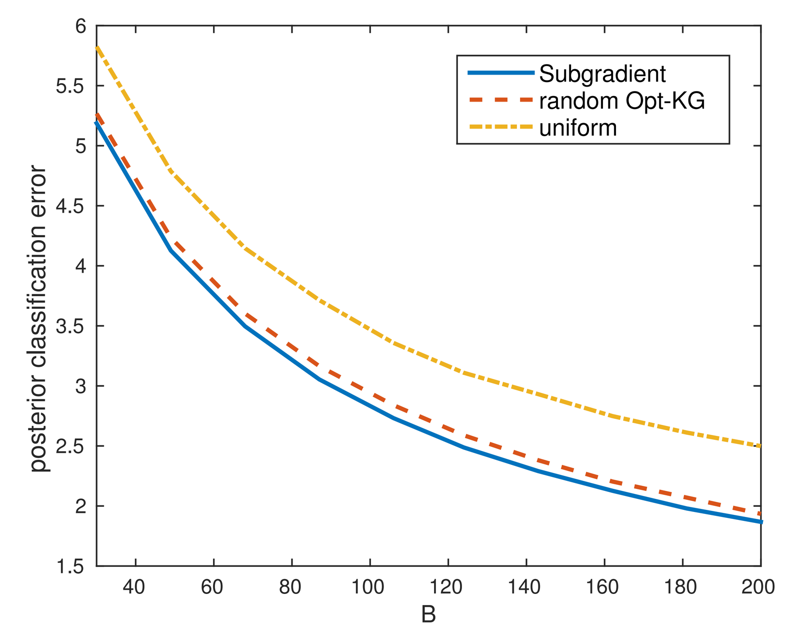
$$\mathcal{E}(\boldsymbol{J}_\theta) = \sum_{T \in \mathcal{T}} \sum_{x \in T} \left(HQ(T)e^{-l_\theta(x)} + P_{\boldsymbol{J}_\theta}(T)\right) |\boldsymbol{J}_\theta(x) - \boldsymbol{L}\boldsymbol{J}_\theta(x)|$$

is related to how far $\boldsymbol{J}_\theta$ is from the optimal value function.

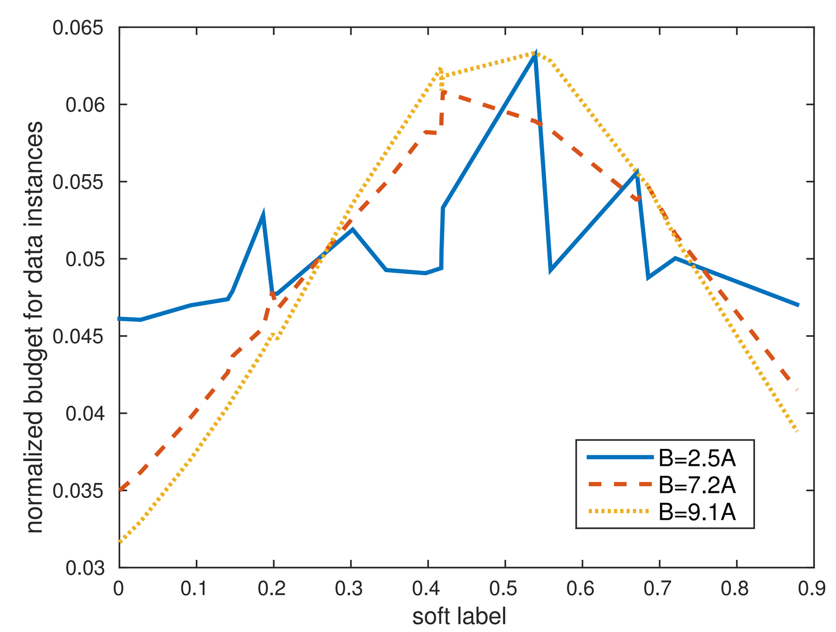
Crowdsourcing

- Need to label A items.
- Each item has soft label $\mu_i \in [0,1]$
- Guess if $\{\mu_i \ge \frac{1}{2}\}$ for as many i as we can
- For $t = 1, \dots, T$:
  - Pick $i \in \{1, \dots, A\}$
  - Recieve $X_t \sim \text{Bern}(\mu_i)$
- Use Beta prior $\Rightarrow$ MDP dynamics equivalent to Bayesian updates
- P_0 limits transitions,
- $q(x)$ rewards correct labels

Experimental Results



Average error of three policies. Our method requires 10% fewer samples for same accuracy



Portion of budget vs. soft label. Harder soft labels receive more budget, and the difference grows with B.