Instance-Optimal PAC Algorithms for Contextual Bandits

Zhaoqi Li*, Lillian Ratliff*, Houssam Nassif†, Kevin Jamieson*, Lalit Jain*

*University of Washington
†Amazon
Contextual Bandit Setting

At each time $t = 1, 2, \cdots$:
- Context $c_t \in \mathcal{C}$ arrives, $c_t \sim \nu \in \Delta_{\mathcal{C}}$
- Choose action $a_t \in \mathcal{A}$
- Receive reward $r_t$, $\mathbb{E}[r_t | c_t, a_t] = r(c_t, a_t) \in \mathbb{R}$

Policy class $\Pi$, each $\pi \in \Pi$, $\pi : \mathcal{C} \to \mathcal{A}$
- Average reward: $V(\pi) := \mathbb{E}_{c \sim \nu}[r(c, \pi(c))]$
- Optimal policy: $\pi^* := \arg \max_{\pi \in \Pi} V(\pi)$

$(\epsilon, \delta) - \text{PAC Guarantee}$

Return $\hat{\pi}$ satisfying, $V(\hat{\pi}) \geq V(\pi^*) - \epsilon$ with probability greater than $1 - \delta$ in a minimum number of samples.

Contributions:
- Show the first instance-dependent lower bound for PAC contextual bandits
- Design sampling procedure that achieves this lower bound
- Design a computationally efficient algorithm - allowing context space $\mathcal{C}$ and policy space $\Pi$ to be infinite!
Regret Minimization Not Enough

- Regret heavily studied:
  - ILOVETOCONBANDITS [Agarwal et al. 2014] achieves $R_T = O(\sqrt{|A| T \log(\Pi)})$, computationally efficient
  - Modification gives $(\epsilon, \delta)$- PAC algorithm w/ sample complexity $O( |A| \log(\Pi/\delta)/\epsilon^2)$, also see [Zanette et al. 2021]

Two Problems

a) Minimax Result! Does not adapt to hardness of instance.

b) Can construct an example, where any optimal regret algorithm won’t be instance optimal!
Agnostic Setting Reduces to Linear

- Lower bound motivated by best-arm identification in linear bandits [Fiez et al. 2019]
- Let $\theta^* \in \mathbb{R}^{|C| \times |A|}$ where $[\theta^*]_{c,a} = r(c, a)$

\[
\begin{pmatrix}
c \\
\vdots \\
r(c, a)
\end{pmatrix}
\xrightarrow{\text{vectorize}}
\theta^*
\]

\[
r(c, a) = \langle \text{vec}(e_c e_a^T), \theta^* \rangle
\]

\[
\phi(c, a)
\]
Contribution 1: A Lower Bound

Let $\phi_{\pi} := \mathbb{E}_{c \sim \nu}[\phi(c, \pi(c))]$ and $A(p) = \sum_{c} \nu_{c} \sum_{a} p_{c,a} \phi(c, a) \phi(c, a)^{\top}$.

Theorem [Li et al. 2022] Let $\tau$ be the stopping time of the algorithm. Any $(0, \delta)$-PAC algorithm satisfies $\tau \geq \rho_{\Pi,0} \log(1/2.4\delta)$ with high probability where

$$\rho_{\Pi,0} = \min_{p_{c} \in \Delta_{A}, \forall c \in C} \max_{\pi \in \Pi \setminus \pi^{*}} \frac{\|\phi_{\pi^{*}} - \phi_{\pi}\|_{A(p)^{-1}}^{2}}{\Delta(\pi)^{2}}.$$  

• This bound is better than the sample complexity bound based on disagreement coefficients [Foster et al. 2020] and decision-estimation coefficients [Foster et al. 2021]
Contribution 2: An Instance-Optimal Algorithm

- In each round, choose $p_c \in \Delta_A, \forall c \in C$ and $n$ such that

$$\min_{p_c \in \Delta_A, \forall c \in C} \max_{\pi \in \Pi} \left( -\Delta(\pi) + \sqrt{\frac{\|\phi_\pi - \phi_{\pi^*}\|_{A(p)}^2 \log(1/\delta)}{n_l}} \right) \leq 2^{-l}$$

Theorem [Li et al. 2022] The algorithm returns an $(\epsilon, \delta)$-PAC policy with at most $O(\rho_{\Pi,\epsilon} \log(|\Pi|/\delta) \log_2(1/\epsilon))$ samples.
Contribution 3: An Efficient Algorithm

- Consider the dual formulation of the design of the previous algorithm:

  $$\textbf{Primal} \quad \min_{p_c \in \Delta_A, \forall c \in C} \max_{\pi \in \Pi} - \Delta(\pi, \pi^*) + \sqrt{\frac{\|\phi_\pi - \phi_{\pi^*}\|_2^2 \log(1/\delta)}{n}}$$

  $$\textbf{Dual} \quad \max_{\lambda \in \Delta_\Pi} \min_{\gamma \geq 0} \min_{p_c \in \Delta_A, \forall c \in C} \sum_{\pi \in \Pi} \lambda_\pi \left( -\Delta(\pi, \pi^*) + \gamma_\pi \|\phi_\pi - \phi_{\pi^*}\|_2^2 + \frac{\log(1/\delta)}{2\gamma_\pi n} \right).$$

  analytical solution ⇒ implicitly maintain $p_c$ for all $c \in C$ simultaneously!

- The dual objective is concave in $\lambda$ and locally strongly convex in $\gamma$, so the saddle point problem can be solved.

- Frank-Wolfe subroutine gives us a sparse yet good enough solution $\lambda$.
Thank you!