A fully adaptive algorithm for pure exploration in linear bandits

Liyuan Xu[1,2], Junya Honda[1,2], Masashi Sugiyama[2,1]

[1]The University of Tokyo [2]RIKEN

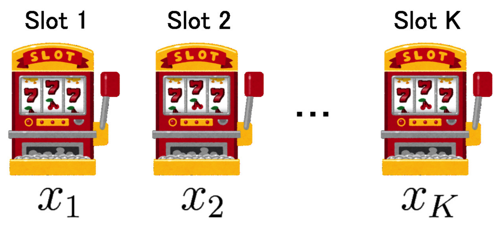
Abstract

- Proposed a new adaptive algorithm for best arm identification in linear bandits, LinGapE (Linear Gap-based Exploration).
- Derived the sample complexity of LinGapE, which matches the sample complexity of an oracle algorithm up to a constant in some limit.
- Showed superiority of LinGapE through experiments based on synthetic and realistic settings.

Problem Settings

Linear Bandits

- The set of arms $[K] = 1, 2, \ldots, K$ and the features of arms $x_1, x_2, \ldots, x_K \in \mathbb{R}^d$
- At each round t, an agent pulls one arm $a_t \in [K]$ and observes reward r_t.
- Rewards r_t is determined as $r_t = x_{a_t}^\top \theta + \varepsilon$.
  - ε: R-sub-Gaussian noise
  - θ: unknown parameter with l_2-norm at most S
- The best arm $a^* = \arg\max_i x_i^\top \theta$



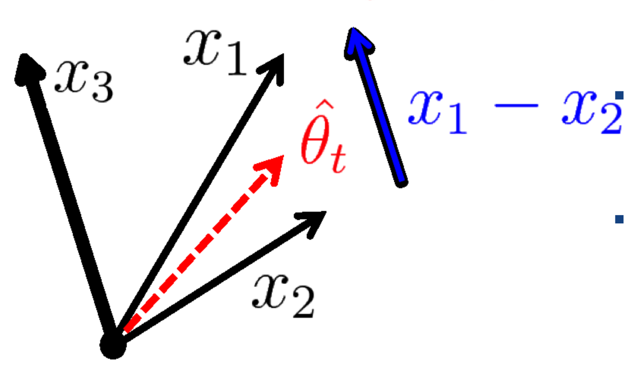
(ε, δ)-Best Arm Identification Problem

Goal: Find an arm $\hat{a}$ satisfying $\mathbb{P}[x_{a^*}^\top \theta - x_{\hat{a}}^\top \theta \geq \varepsilon] \leq \delta$ within a small number of rounds.

$\rightarrow$ Need to design an arm selection strategy and a stopping condition.

Applications: Optimizing sensor network, automatic parameter tuning [2]

Characteristic: Pulling sub-optimal arms can lead to efficient exploration.



Under the current estimation $\hat{\theta}_t$ of θ, arms 1 and 2 have high expected rewards.

- We can directly estimate the gap between these expected rewards by pulling arm 3.

Confidence Bounds

There are two types of the confidence bounds on θ for sequence of arm selection $\mathbf{x}_n = (x_{a_1}, \ldots, x_{a_n})$ and $A_{\mathbf{x}_n} = \sum_{t=1}^n x_{a_t} x_{a_t}^\top$, $b_{\mathbf{x}_n} = \sum_{t=1}^n r_t x_{a_t}$.

Confidence Bound for Static Strategies [3]

For any fixed sequence $\mathbf{x}_n$, if noise variable ε is bounded $\varepsilon \in [-R, R]$, (which is R-sub-Gaussian)

$$|x^\top \theta - x^\top \hat{\theta}_n| \leq 2R \|x\|_{A_{\mathbf{x}_n}^{-1}} \sqrt{2 \log (6n^2 K / (\delta \pi^2))}, \quad \hat{\theta}_n = A_{\mathbf{x}_n}^{-1} b_{\mathbf{x}_n} \tag{1}$$

holds for all $n \in \mathbb{N}$ and all $x \in \{x_i\}_{i=1}^K$ with probability at least $1 - \delta$ for $\|x\|_A = \sqrt{x^\top A x}$.

Confidence Bound for Adaptive Strategies [1]

For any arm selection sequence $\mathbf{x}_n$ and $A_{\mathbf{x}_n}^\lambda = \lambda I + A_{\mathbf{x}_n}$ for $\lambda > 0$,

$$|x^\top \theta - x^\top \hat{\theta}_n^\lambda| \leq R \|x\|_{(A_{\mathbf{x}_n}^\lambda)^{-1}} \sqrt{2 \log (\det(A_{\mathbf{x}_n}^\lambda)^{\frac{1}{2}} K / (\lambda^{\frac{d}{2}} \delta))} + \lambda^{\frac{1}{2}} S, \quad \hat{\theta}_n^\lambda = (A_{\mathbf{x}_n}^\lambda)^{-1} b_{\mathbf{x}_n} \tag{2}$$

holds for all $n \in \mathbb{N}$ and all $x \in \{x_i\}_{i=1}^K$ with probability at least $1 - \delta$.

(2) is valid for adaptive strategies, but looser by $\sqrt{\log(\det(A_{\mathbf{x}_n}^\lambda))} = O(\sqrt{d})$.

Prior Methods

Work by Soare et al. [3]

- Constructs a stopping condition based on (1) to avoid $O(\sqrt{d})$ looseness of (2).
- Proposes static and semi-adaptive arm selection strategies which make (1) valid.
- Derives the lower bound of sample complexity for static strategies.

Arm selection strategies:

- $\mathcal{XY}$-static: Fix all arm selection before observing any samples.
  - Arm selection strategy based on the literature of transductive experimental design.
  - Cannot change arm selection adaptively based on rewards.
- $\mathcal{XY}$-adaptive: Semi-adaptive algorithm that adaptively changes static arm allocations.
  - Divide rounds into multiple phases, employ different arm allocations in different phases.
  - Must discard all samples collected in previous phases for the validity of (1).

Lower bound of static strategies:

- The lower bound is $\Omega(H^{\text{oracle}} \log 1/\delta)$, where H^{oracle} is defined as

$$H^{\text{oracle}} = \min_{\{p_k\}_{k \in [K]}} \max_{i \in [K] \backslash \{a^*\}} \frac{\|x_{a^*} - x_i\|_{\Lambda_p^{-1}}^2}{\Delta_i^2} \quad \text{s.t.} \sum_{k=1}^K p_k = 1,\ p_k \geq 0,\ \Lambda_p = \sum_{k=1}^K p_k x_k x_k^\top \tag{3}$$

for $\Delta_i = x_{a^*}^\top \theta - x_i^\top \theta$.

- Lower bound is derived by considering an oracle algorithm, $\mathcal{XY}$-oracle.
- $\mathcal{XY}$-oracle computes the optimal arm selection ratio using true θ, which is unknown in reality.

Our Contributions

- Use a stopping condition based on (2), which allows employing adaptive strategies.
- Prove that $O(\sqrt{d})$ looseness in (2) does not appear in the main term of the sample complexity.
- Confirm that looseness of (2) does not harm performances empirically.

Proposed Method: LinGapE

LinGapE = Linear Gap-based Exploration

Stopping condition:

- Based on the confidence bound in (2).
- Valid for adaptive strategies as well.

Arm selection strategy:

At each round t, repeat the following.

- Nominate two arms i_t, j_t.
- Pull the arm a_t that discriminates i_t, j_t the most.

Algorithm 1: LinGapE

Get an initial estimation $\hat{\theta}_K$ by pulling each arm once.;

Loop $t = K, K+1, \ldots$

- // Nominate (i_t, j_t) for candidates
- $i_t, j_t, B(t) \leftarrow$ Select-direction($\hat{\theta}_t$);
- **if** $B(t) \leq \varepsilon$ **then**
  - Return i_t as the best arm $\hat{a}^*$;
- // Pull arms for estimating the gap of them
- Select the arm a_{t+1} based on (5);
- Pull arm a_{t+1} and update estimation $\hat{\theta}_{t+1}$;

The algorithm for nominating i_t, j_t

Arm i_t is the estimated best arm, and arm j_t is the arm that is the most likely to surpass i_t.

Algorithm 2: Select-direction

Procedure *Select-direction*($\hat{\theta}_t$):

- $i_t \leftarrow \arg\max_{i \in [K]} (x_i^\top \hat{\theta}_t)$;
- $j_t \leftarrow \arg\max_{j \in [K]} (\hat{\Delta}_t(j, i_t) + \beta_t(j, i_t))$;
- $B(t) \leftarrow \max_{j \in [K]} (\hat{\Delta}_t(j, i_t) + \beta_t(j, i_t))$;
- Return i_t, j_t, $B(t)$;

$$\beta_t(i, j) = \|x_i - x_j\|_{(A_t^\lambda)^{-1}} \left(R \sqrt{2 \log \frac{\det(A_t^\lambda)^{\frac{1}{2}} \det(\lambda I)^{-\frac{1}{2}}}{\delta}} + \lambda^{\frac{1}{2}} S \right)$$

$$\hat{\Delta}_t(i, j) = (x_i - x_j)^\top \hat{\theta}_t, \quad A_t^\lambda = \lambda I + \sum_{k=1}^t x_{a_k} x_{a_k}^\top$$

The algorithm for selecting a_{t+1}

- Compute the optimal arm selection ratio $\{p_k^*(i, j)\}_{k \in [K]}$ for discriminating arms i and j by

$$\{p_k^*(i, j)\}_{k \in [K]} = \arg\min_{\{p_k\}_{k \in [K]}} \|x_i - x_j\|_{\Lambda_p^{-1}}^2 \quad \text{s.t.} \sum_{k=1}^K p_k = 1,\ p_k \geq 0,\ \Lambda_p = \sum_{k=1}^K p_k x_k x_k^\top, \tag{4}$$

which can be solved by the linear program.

- a_{t+1} is decided based on i_t, j_t as follows.

$$a_{t+1} = \arg\min_{a \in [K]: p_a^*(i_t, j_t) > 0} T_a(t) / p_a^*(i_t, j_t), \tag{5}$$

where $T_a(t)$ is the number of times that arm a is pulled until round t.

Theoretical Analysis

Theorem 1: Sample Complexity of LinGapE

If $\lambda \leq \frac{2R^2}{S^2} \log \frac{K^2}{\delta}$, then the number of samples τ of LinGapE satisfies

$$\mathbb{P}\left[\tau \leq 8 H_\varepsilon R^2 \log \frac{K^2}{\delta} + C(H_\varepsilon, \delta)\right] \geq 1 - \delta, \quad H_\varepsilon = \sum_{k=1}^K \max_{i, j \in [K]} \frac{p_k^*(i, j) \rho(i, j)}{\max \left(\varepsilon, \frac{\varepsilon + \Delta_i}{3}, \frac{\varepsilon + \Delta_j}{3}\right)^2},$$

where $C(H_\varepsilon, \delta) = O\left(d H_\varepsilon \log \left(H_\varepsilon \log \frac{1}{\delta}\right)\right)$ and $\rho(i, j)$ is the optimal value of (4).

As shown above, looseness of $O(\sqrt{d})$ does not affect the main term. Furthermore, the following statements holds for H^{oracle} in (3):

$$H_\varepsilon \leq 72 K H^{\text{oracle}}, \quad H_\varepsilon \to 72 H^{\text{oracle}} \ (\Delta_1 / \Delta_i \to 0).$$

The performance of LinGapE matches the oracle algorithm in this limit.

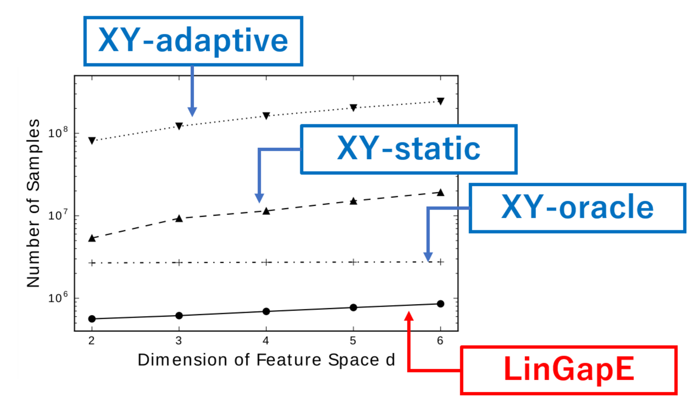
Experiments

Synthetic setting used in [3]:

- The number of arms is $K = d + 1$, where features are

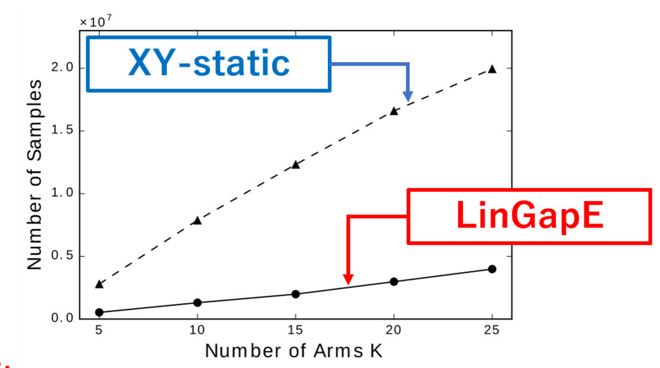
$$x_1 = e_1,\ x_2 = e_2,\ \ldots,\ x_d = e_d,\ x_{d+1} = (\cos(0.01), \sin(0.01), 0, \ldots, 0)^\top.$$

- Set $\theta = (2, 0, \ldots, 0)^\top$.
  - $x_1^\top \theta = 2$ vs. $x_{d+1}^\top \theta = 2\cos(0.01) \approx 1.9999$.
  - Arm 2 can discriminate arms 1 and $d + 1$.
- LinGapE mostly select arm 2.
- Thus, $\det(A_{\mathbf{x}_n}^\lambda) = o(n^d)$, which makes (2) tight.
- LinGapE stops faster than $\mathcal{XY}$-oracle.



Setting based on Yahoo! Webscope Dataset R6A [4]:

- Consists of pairs of user-article feature x and target y ($y = 1$ if seen, and $y = 0$ otherwise).
- Relatively high-dimensional data (36-dimensional).
- Estimate θ by linear ridge regression in $y = x^\top \theta$.
- Run simulations based on the estimated θ.
- 5 times less observations compared to $\mathcal{XY}$-static.
- Less dependent on K compared to $\mathcal{XY}$-static.
- Looseness of (2) does not harm empirical performances.



[1] Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. NIPS2011.

[2] M. Hoffman, B. Shahriari, and N. Freitas. On correlation and budget constraints in model-based bandit optimization with application to automatic machine learning. AISTATS2014.

[3] M. Soare, A. Lazaric, and R. Munos. Best-arm identification in linear bandits. NIPS2014.

[4] https://webscope.sandbox.yahoo.com/

THE UNIVERSITY OF TOKYO

RIKEN