Piecewise-stationary Bandit Problems with Side Observations

Jia Yuan Yu[1] Shie Mannor[1 2]

[1]Department of Electrical and Computer Engineering
McGill University

[2]Faculty of Electrical Engineering
Technion

COLT Workshops, Montréal. June 2009

Multiarmed bandits

- Stochastic multi-armed bandit:
  - no change-point.
- Adversarial multi-armed bandit:
  - arbitrarily many change-points.

Our model: somewhere in between.

Multiarmed bandits

- Stochastic multi-armed bandit:
  - no change-point.
- Adversarial multi-armed bandit:
  - arbitrarily many change-points.

Our model: somewhere in between.

Example: Piecewise-stationary MAB

- Arms: $1, \ldots, n$.
- Each arm has a piecewise-stationary reward distribution.
  - Stationary distribution, but with abrupt changes at unknown instants.
- Query arm i: ask gambler at arm i.

Example 2: Insurance policies

- Arms: different policies.
- Reward: sales.
- Queries: surveys.

The problem

- At each time instant:
  - choose and follow one expert,
  - query the reward of other experts,
  - receive reward of chosen expert,
  - observe rewards of queried experts,
  - pay cost-of-query.
- Goal: do as well as if reward distributions and change-points were known in advance.

Notation: Piecewise-stationary rewards

- Experts $1, \ldots, n$.
- $b_t(i) \in [0, 1]$ is reward of i-th expert at time t
- Rewards form a sequence of random vectors:

$$\underbrace{\begin{bmatrix} b_1(1) \\ b_1(2) \\ \vdots \\ b_1(n) \end{bmatrix}, \ldots, \begin{bmatrix} b_{\nu_2}(1) \\ b_{\nu_2}(2) \\ \vdots \\ b_{\nu_2}(n) \end{bmatrix}}_{\text{distribution } f_1,\text{ mean } \beta_1}, \quad \cdots \quad \underbrace{\begin{bmatrix} b_{\nu_j}(1) \\ b_{\nu_j}(2) \\ \vdots \\ b_{\nu_j}(n) \end{bmatrix}, \ldots, \begin{bmatrix} b_{\nu_{j+1}}(1) \\ b_{\nu_{j+1}}(2) \\ \vdots \\ b_{\nu_{j+1}}(n) \end{bmatrix}}_{\text{distribution } f_j,\text{ mean } \beta_j}, \quad \cdots$$

- Unknown change-points: $\nu_2, \nu_3, \ldots$
- Between change-points: fixed, but unknown distributions $f_1, f_2, \ldots$

More notations

- At each time step t,
  - pick one expert, say a_t,
  - query the reward of a subset of additional experts, say $\mathcal{S}_t$,
  - receive reward $b_t(a_t)$,
  - observe $\{b_t(j) \text{ for } j \in \mathcal{S}_t\}$,
  - pay query cost $C_Q(|\mathcal{S}_t|)$.

Objective

- Optimal expected reward: when every reward distribution is known in advance.
- Expected regret at time T:

$$R_T \triangleq \sum_{t=1}^{T} \max_{i=1,\ldots,n} \beta_t(i) - \sum_{t=1}^{T} \mathbb{E}[b_t(a_t)]. \tag{1}$$

- If we make ℓ queries per step, the overall cost-per-step is

$$R_T/T + C_Q(\ell).$$

Known results

Stochastic MAB:

- Expected regret of $O(n\log(T))$.

Adversarial MAB:

- Different notion of adversarial regret:

$$\max_{i=1,\dots,n}\sum_{t=1}^{T}\beta_t(i)-\sum_{t=1}^{T}\mathbb{E}[b_t(a_t)]. \tag{2}$$

- Regret of $O(\sqrt{Tn\log(n)})$.

Known results for piecewise-stationary bandit without queries

- (Hartland et al., 2006) provide a partial solution that detects only distribution changes in the current best expert.

Definition

$k \triangleq$ number of changes up to time T

- If *k is known in advance*:
  - Fixed-share algorithm (Herbster & Warmuth, 1998) gives a regret of $O(\sqrt{nkT \log(T)})$.
  - Discounted- & Sliding-window-UCB algorithm (Garivier & Moulines, 2008) give a regret of $O(n\sqrt{kT} \log(T))$.
  - Lower-bound of $\Omega(\sqrt{T})$ without queries.

Our result for piecewise-stationary rewards

- Queries reduce regret from $O(n\sqrt{kT}\log(T))$ to $O(nk\log(T))$,
- without prior knowledge of k.

Our approach

- Run standard algorithm for stochastic MAB,
  - e.g., UCB algorithm of (Auer et al., 2002a).
- Reset when we detect a change-point.

- How to detect change-points?
  - Naive solution: detect changes in the distribution directly using change-detection algorithms (CUSUM, etc.).
  - Simpler solution: detect changes in the empirical mean over windows of appropriate length.

Our approach

- Run standard algorithm for stochastic MAB,
  - e.g., UCB algorithm of (Auer et al., 2002a).
- Reset when we detect a change-point.
- How to detect change-points?
  - Naive solution: detect changes in the distribution directly using change-detection algorithms (CUSUM, etc.).
  - Simpler solution: detect changes in the empirical mean over windows of appropriate length.

Our approach

- Run standard algorithm for stochastic MAB,
  - e.g., UCB algorithm of (Auer et al., 2002a).
- Reset when we detect a change-point.
- How to detect change-points?
  - Naive solution: detect changes in the distribution directly using change-detection algorithms (CUSUM, etc.).
  - Simpler solution: detect changes in the empirical mean over windows of appropriate length.

WMD algorithm

Windowed Mean-shift Detection

Break time horizon into intervals of length τ, compute empirical mean in each interval:

$$\underbrace{b_1, b_2, \ldots, b_\tau}_{\hat{b}_1}, \underbrace{b_{\tau+1}, \ldots, b_{2\tau}}_{\hat{b}_2}, \ldots, \underbrace{b_{(m-1)\tau+1}, \ldots, b_{m\tau}}_{\hat{b}_m} \ldots$$

At each time step t:

1. Follow UCB algorithm:
   - Play the expert with highest upper-confidence index:

$$\hat{b}(i) + \sqrt{2\log(T)/\#(i)}.$$

2. Query experts with equal frequency.
3. Detect changes: If $\left\|\hat{b}_m - \hat{b}_r\right\|_\infty > \epsilon$, reset UCB sub-algorithm.

Guarantee

Piecewise-stationary bandit with queries

Theorem

Suppose that at every change-point, the mean reward of some expert changes by at least 2ϵ*, i.e.,* $\left|\beta_{\nu_j}(i) - \beta_{\nu_{j+1}}(i)\right| > 2\epsilon$*. The WMD algorithm with windows of length* $\tau = \lfloor \frac{n}{\ell} \rfloor \cdot \lfloor \frac{\log(T)}{2\epsilon^2} \rfloor$ *achieves a regret of*

$$R_T \leq \frac{7}{\epsilon^2} \frac{kn}{\ell} \log(T) + \frac{C}{\Delta^2} kn \log(T) + \frac{6C}{\Delta^2} n^2, \tag{3}$$

for every sequence of change-points $\nu_1, \nu_2, \ldots$ *and every choice of post-change distributions* $f_{\nu_1}, f_{\nu_2}, \ldots$

- This regret is $O(nk \log(T))$ without prior knowledge of k, but with queries.
- Recall: Discounted- and Sliding-window UCB (Garivier & Moulines, 2008) have regret of $O(n\sqrt{kT} \log(T))$ with prior knowledge of k, but without queries.

Proof ideas

- $L \triangleq$ Expected number of intervals between change-point and its detection.
- $N(T) \triangleq$ Expected number of false detections up to T.

- Two components of the regret:
  - expected number of resets is $k + N(T)$; hence, regret between resets is

$$(Cn/\Delta^2)(k + N(T)) \log(T).$$

  - regret due to delay in detection:

$$k(L + 1)\tau.$$

- Bound $N(T)$ and L using Hoeffding's Inequality on empirical mean of i.i.d. rewards in each window.

A (partial) lower bound

- Consider class of algorithms that detect-and-react.
  - Constraint: as many switches between experts as distribution changes;
  - or a vanishing frequency of false detection.
- Optimal delay to change-detection is $\Omega(\log(T))$ (Lorden, 1971).
- For such algorithms, regret is lower-bounded by $\Omega(k \log(T))$.

Shiryayev-Roberts scheme

Raise alarm ($f_0 \to f_\theta$) at time t if

$$\sum_{k=1}^{t} \frac{f_\theta(b_k)\dots f_\theta(b_t)}{f_0(b_k)\dots f_0(b_t)} \geq A. \tag{4}$$

Theorem (Average run-lengths (Pollak, 1985))

1. *If there is no change, then* $\mathbb{E}_\infty[\mathit{Alarm\ Time}] \geq A$.
2. *If a change occurs at time* 1*, then*

$$\mathbb{E}_1[\mathit{Alarm\ Time}] = [\log A + \log\log A + O(1)]/D(f_\theta; f_0).$$

This is optimal.

Detect-and-act algorithms

Class of algorithms where:

- $\#$ swtiches between experts $\leq \#$ distribution changes $+1$

Consequence:

- Expected $\#$ false-detections ≤ 1.

Theorem (Regret lower-bound)

For every fixed algorithm of our class (fixed ℓ and query mechanism), there exists a piecewise-stationary source such that detect-and-react algorithm has a regret of at least

$$R_T \geq k\frac{n}{\ell}\log T / D(f_\theta; f_0).$$

Proof sketch:

- Bernoulli-distributed sources:
  - Detecting change in distribution requires detecting change in mean.
- Average run-length theorem.

Query-regret trade-off

- Overall expected cost-per-step at time T:

$$C_Q(\ell) + R_T/T.$$

- If $C_Q(\ell) = c_q \times \ell$, then optimize cost-per-step with respect to ℓ.
- Optimal query rate is

$$\ell^* = \sqrt{(7kn/c_q)\log(T)/T}.$$

Open questions

- Non-uniform querying (confidence bound-type algorithms for mean-shift detection).
- Probabilistic queries (with a failure probability).
- Lower bound on queries to detect mean-shift.
- Restless bandit problems.

References

Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002a). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, *47*, 235–256.

Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R. E. (2002b). The nonstochastic multiarmed bandit problem. *SIAM J. Computing*, *32*, 48–77.

Garivier, A., & Moulines, E. (2008). On upper-confidence bound policies for non-stationary bandit problems. Preprint. `http://arxiv.org/abs/0805.3415`.

Hartland, C., Gelly, S., Baskiotis, N., Teytaud, O., & Sebag, M. (2006). Multi-armed bandit, dynamic environments and meta-bandits. Preprint. `http://hal.archives-ouvertes.fr/hal-00113668/en/`.

Herbster, M., & Warmuth, M. K. (1998). Tracking the best expert. *Machine Learning*, *32*, 151–178.

Kocsis, L., & Szepesvári, C. (2006). Discounted-UCB. 2nd PASCAL Challenges Workshop.

Lorden, G. (1971). Procedures for reacting to a change in distribution. *Ann. Math. Statist.*, *42*, 1897âĂŞ–1908.

Pollak, M. (1985). Optimal detection of a change in distribution. *Ann. Statist.*, *13*, 206–227.