Contextual Bandits with Similarity Information

Alex Slivkins
Microsoft Research Silicon Valley

COLT 2011
Running example

- **Web advertisement.**
  Every time someone visits one of your web pages, you display an ad. There are many ads to choose from. To goal is to maximize #clicks.
  - you can update your selection based on the clicks received
Multi-Armed Bandits

- In a (basic) MAB problem one has:
  - set $Y$ of alternatives (a.k.a. *arms*)
  - $\mu(y) \in [0,1]$ expected payoff for each $y \in Y$ (fixed, unknown)
- In each round: algorithm picks arm $y \in Y$ based on past history
  - payoff: independent sample in $[0,1]$ with expectation $\mu(y)$
- Classical setting that models exploration-exploitation trade-off

- Web ads
  - arms $\rightarrow$ ads
  - payoffs $\rightarrow$ clicks

$\mu = .6$  $\mu = .2$  $\mu = .4$
Contextual Bandits

- In each round:
  - context $x \in X$ arrives
  - algorithm picks arm $y \in Y$
  - payoff: independent sample in $[0,1]$ with expectation $\mu(x,y)$

- A problem instance contains: set $X$ of contexts, set $Y$ of arms, 
  $\mu: X \times Y \rightarrow [0,1]$ expected payoffs (fixed but unknown)

Web ads
contexts $\rightarrow$ pages
arms $\rightarrow$ ads
payoffs $\rightarrow$ clicks
Similarity info

• Similarity between contexts: a metric $D_X$ on contexts s.t.

$$| \mu(x, y) - \mu(x', y) | \leq D_X(x, x')$$

• allows to generalize from one context to another

• Similarity between arms: a metric $D_Y$ on arms s.t.

$$| \mu(x, y) - \mu(x, y') | \leq D_Y(y, y')$$

• esp. useful if #arms is very large or infinite

\[ \begin{array}{c}
\text{contexts } X \\
\hline
x \hline
\text{arms } Y
\end{array} \]
More generally, an algorithm is given similarity distance on context-arm pairs: a metric $D$ such that

$$| \mu(x,y) - \mu(x',y') | \leq D((x,y), (x',y'))$$

"similarity space" $(X \times Y, D)$
Problem formulation

- Problem instance:
  - Revealed to the algorithm: similarity space \((X \times Y, D)\)
  - Fixed but not revealed: \(\mu: X \times Y \to [0,1]\) expected payoffs, a sequence of context arrivals \(x_1, x_2, x_2, \ldots\)
- Promise: \(|\mu(x, y) - \mu(x', y')| \leq D((x, y), (x', y'))\)
- Context-specific benchmark: \(\mu^*(x) = \max_{y \in Y} \mu(x, y)\)

**Contextual Regret**: \(\sum_t \mu^*(x_t) - \mu(x_t, y_t)\)
Some background

- Context-free MAB with stochastic payoffs: well-understood

- Contextual bandits
Prior work: uniform partitions

- Partition contexts into “sub-intervals” of the same size, treating each sub-interval as a single context; same for arms.
- Once the partition is chosen, similarity is ignored. After some time, take a finer partition (but still uniform).
- The partitions do not depend on observations.

Pros: Generalizes across similar contexts and across similar arms. Regret bounds are worst-case optimal.
Prior work: uniform partitions

**Drawbacks**

- refines “rectangles” with low payoffs or infrequent context arrivals, which is wasteful for exploration.
- does not take advantage of “nice” problem instances

![Diagram showing contexts and arms with different payoffs](image)
This paper: adaptive partitions

- Idea: adapt the partition to observed payoffs & contexts. **Refine the partition only where it matters:** in regions with high payoffs and frequent context arrivals.

- Result: (optimal) instance-dependent guarantees that capture the “goodness” of a problem instance

Done for context-free setting in R. Kleinberg, Slivkins, Upfal (STOC’08), Bubeck et al (NIPS’08)
Rest of the talk

- Algorithm and guarantees
- Unexpected applications
- An algorithm for adversarial payoffs
Algorithm: contextual zooming

- Similarity space is covered by active balls
- For each active ball, define a number (index)
- In each round: pick relevant active balls with largest index, pick an arm which intersects this ball

Index of ball B is a UCB
\[
\text{Index}(B) \geq \mu(x,y) \quad \text{for any } (x,y) \in B
\]

Index(B) depends on:
- payoff(B), radius(B),
- #samples(B)
Algorithm: contextual zooming

- Refining the partition ("zooming in"): if some active ball $B$ of radius $r$ has been selected $1/r^2$ times, activate several balls of radius $r/2$ that cover $B$

- Intuition: $B$ selected often $\Rightarrow$ frequent context arrivals and high payoffs $\Rightarrow$ worth taking a closer look

E.g. we define $\text{Index}(B)$ as $\min_B \text{ "PreIndex" } (B') + D(B,B')$ $B'$: active balls of $\geq$ radius
Unexpected application: slowly changing payoffs

- expected payoff of each arm changes by \( \leq \varepsilon \) in each round
- **idea**: treat time \( t \) (current round) as a context
  \[
  | \mu(t, y) - \mu(t', y) | \leq \varepsilon | t - t' |
  \]
- algorithm: contextual zooming, restart every \( N \) rounds
  optimal contextual regret (averaged over time)

In the paper: more on slowly changing payoffs
E.g. we recover & generalize a result from
Slivkins & Upfal (COLT 2008)
Unexpected application: sleeping bandits

- In each round, some arms are “asleep” (not available)
- **idea**: context = set of “awake” arms
  - not all context-arm pairs are feasible
  - distance(pair1, pair2) = 1 (no restriction whatsoever)
- Recover & generalize a result from R. Kleinberg et al (COLT 2008)
Other results

- Another application: online learning-to-rank
  (Slivkins, Radlinski, Gollapudi ICML 2010)
  - contextual zooming is a subroutine;
  (again) contexts mean something different
- Adversarial payoffs (this paper)
  - “meta-algorithm”: plug in your favorite bandit algorithm, adaptively refine the space of contexts.