Bandits with Knapsack

Lixing Lyu

Institute of Operations Research and Analytics,
National University of Singapore

August 5, 2022

Motivation (Simple example in dynamic pricing)

- Suppose we are going to sell 20 apples.

Motivation (Simple example in dynamic pricing)

- Suppose we are going to sell 20 apples.
- At each round $t = 1, 2, ..., T$,
  1. Set price p_t for the apple.
  2. Customer decides to buy n_t apples.
  3. Revenue $= n_t p_t$.
  4. If all apples are sold, **STOP**!

Motivation (Simple example in dynamic pricing)

- Suppose we are going to sell 20 apples.
- At each round $t = 1, 2, ..., T$,
  1. Set price p_t for the apple.
  2. Customer decides to buy n_t apples.
  3. Revenue $= n_t p_t$.
  4. If all apples are sold, **STOP**!
- Goal: Maximize total Revenue.

Motivation (Simple example in dynamic pricing)

- Suppose we are going to sell 20 apples.
- At each round $t = 1, 2, ..., T$,
  1. Set price p_t for the apple.
  2. Customer decides to buy n_t apples.
  3. Revenue $= n_t p_t$.
  4. If all apples are sold, **STOP**!
- Goal: Maximize total Revenue.
- Decision making: Appropriately set price in each round.

Motivation (Simple example in dynamic pricing)

- Suppose we are going to sell 20 apples.
- At each round $t = 1, 2, ..., T$,
  1. Set price p_t for the apple.
  2. Customer decides to buy n_t apples.
  3. Revenue $= n_t p_t$.
  4. If all apples are sold, **STOP**!
- Goal: Maximize total Revenue.
- Decision making: Appropriately set price in each round.
- Challenge: We only have 20 apples.

Other Application

- Dynamic pricing with limited supply.
- Advertisement allocation with limited budget.
- Auction.
- Crowdsourcing.
- Parameter tuning.
- Network design.
- ...

All are the same:

$$\max \quad \text{Revenue/Reward/...} \quad \text{s.t.} \quad \text{Inventory constraints.}$$

Problem setting of Bandit with Knapsack (BwK)

- Horizon: T time periods, which is known.
- Bandits (Actions): K arms. Denote arm set as $\mathcal{A}$.
- Recource: d kinds of resource with budgets $B_1, ..., B_d \leq T$. wlog, set $B_1 = \cdots = B_d = B$.

Problem setting of Bandit with Knapsack (BwK)

- Horizon: T time periods, which is known.
- Bandits (Actions): K arms. Denote arm set as $\mathcal{A}$.
- Recource: d kinds of resource with budgets $B_1, ..., B_d \leq T$. wlog, set $B_1 = \cdots = B_d = B$.
- At each round $t = 1, 2, ..., T$,
    1. Select an arm $A_t \in \mathcal{A}$.
    2. Receive reward $r_t = r_t(A_t) \in [0, 1]$.
    3. Consume $c_t^{(i)} = c_t^{(i)}(A_t) \in [0, 1]$ amount of resource i, $\forall i \in [d]$.
    4. If some budget constraints are violated, then **STOP**.

Problem setting of Bandit with Knapsack (BwK)

- Horizon: T time periods, which is known.
- Bandits (Actions): K arms. Denote arm set as $\mathcal{A}$.
- Recource: d kinds of resource with budgets $B_1, ..., B_d \leq T$. wlog, set $B_1 = \cdots = B_d = B$.
- At each round $t = 1, 2, ..., T$,
  1. Select an arm $A_t \in \mathcal{A}$.
  2. Receive reward $r_t = r_t(A_t) \in [0, 1]$.
  3. Consume $c_t^{(i)} = c_t^{(i)}(A_t) \in [0, 1]$ amount of resource i, $\forall i \in [d]$.
  4. If some budget constraints are violated, then **STOP**.
- Goal: Maximize the total reward.

Problem setting of Bandit with Knapsack (BwK)

- Horizon: T time periods, which is known.
- Bandits (Actions): K arms. Denote arm set as $\mathcal{A}$.
- Recource: d kinds of resource with budgets $B_1, ..., B_d \leq T$. wlog, set $B_1 = \cdots = B_d = B$.
- At each round $t = 1, 2, ..., T$,
  1. Select an arm $A_t \in \mathcal{A}$.
  2. Receive reward $r_t = r_t(A_t) \in [0, 1]$.
  3. Consume $c_t^{(i)} = c_t^{(i)}(A_t) \in [0, 1]$ amount of resource i, $\forall i \in [d]$.
  4. If some budget constraints are violated, then **STOP**.
- Goal: Maximize the total reward.
- Stochastic BwK: $r_t(A_t)$, $c_t^{(i)}(A_t)$ are i.i.d $\sim P_{A_t}$, which is unknown. Denote $\mathbb{E}[P_a] = (r(a), c^{(i)}(a))$ as the expected outcome for all $a \in \mathcal{A}$.
- Adversarial BwK: $r_t(A_t)$, $c_t^{(i)}(A_t)$ are adversarial.

Prior Work

- General model and optimal solutions for Stochastic BwK: Badanidiyuru et al. [2013].
- Some extention: Agrawal and Devanur [2014] (concave reward), Immorlica et al. [2019] (adversarial BwK), Agrawal et al. [2016] (contextual bandit), Sankararaman and Slivkins [2018] (combinatorial semi-bandit),...

,

Main Challenge of BwK: Why BwK hard?

1. Distribution over arms **beats** any fixed arms. See the following example...
   - Suppose $K = 2$, $d = 2$, $B = \frac{T}{2}$,
   - Arm 1 receives reward 1 and consumes 1 unit of resource 1.
   - Arm 2 receives reward 1 and consumes 1 unit of resource 2.
   - For any fixed arm, reward $= \frac{T}{2}$,
   - Each arm with prob. 50%, reward $= T$.

Main Challenge of BwK: Why BwK hard?

1. Distribution over arms **beats** any fixed arms. See the following example...
   - Suppose $K = 2$, $d = 2$, $B = \frac{T}{2}$,
   - Arm 1 receives reward 1 and consumes 1 unit of resource 1.
   - Arm 2 receives reward 1 and consumes 1 unit of resource 2.
   - For any fixed arm, reward $= \frac{T}{2}$,
   - Each arm with prob. 50%, reward $= T$.
2. The exploration phase also consumes resource.

Benchmark

There are totally 3 kinds of benchmark.

1. Best fixed arm in hindsight (Classical bandit problem).

Benchmark

There are totally 3 kinds of benchmark.

1. Best fixed arm in hindsight (Classical bandit problem).
2. Best fixed distribution over arms (Based on LP relaxation).

Benchmark

There are totally 3 kinds of benchmark.

1. Best fixed arm in hindsight (Classical bandit problem).
2. Best fixed distribution over arms (Based on LP relaxation).
3. Best dynamic policy.

Benchmark

There are totally 3 kinds of benchmark.

1. Best fixed arm in hindsight (Classical bandit problem).
2. Best fixed distribution over arms (Based on LP relaxation).
3. Best dynamic policy.

We always use 2 as the benchmark. Denote the expected total reward as OPT.

Performance Metric: Regret

Denote τ be the stopping time when at least one resource has been depleted, then the regret

$$\text{Regret} = \text{OPT} - \mathbb{E}\left[\sum_{t=1}^{\tau} r_t\right] \tag{1}$$

BwK with Adversarial Scaling (Work in progress...)

- Joint work with Prof. CHEUNG Wang Chi.
- K arms, d resource, T time periods.
- At each round $t = 1, 2, ..., T$,
  1. Receive adversarial term $q_t > 0$.
  2. Select an arm $A_t \in \mathcal{A}$.
  3. Receive reward $q_t r_t$.
  4. Consume $q_t c_t^{(i)}$ amount of resource i, $\forall i \in [d]$.
  5. If some budget constraints are violated, then **STOP**.
- $r_t, c_t^{(i)}$ are i.i.d.
- Goal: Maximize the total reward.

Benchmark and Regret I

The Benchmark is defined by the following LP relaxtion model:

$$
\begin{aligned}
\mathrm{OPT} = \max \quad & \sum_{t=1}^{T} q_t \sum_{a \in \mathcal{A}} x_{t,a} r(a) \\
\text{s.t.} \quad & \sum_{t=1}^{T} q_t \sum_{a \in \mathcal{A}} x_{t,a} c^{(i)}(a) \leq B, \ \forall i \in [d] \\
& \sum_{a \in \mathcal{A}} x_{t,a} = 1, \ \forall t \in [T] \\
& x_{t,a} \geq 0, \ \forall t \in [T], a \in \mathcal{A}
\end{aligned}
\tag{2}
$$

Benchmark and Regret II

Denote $u_a = \sum_{t=1}^{T} q_t x_{t,a} / \sum_{t=1}^{T} q_t$, then the problem can be reformulated as

$$\begin{aligned} \mathrm{OPT} = \max \quad & \left(\sum_{t=1}^{T} q_t\right) \cdot \sum_{a\in\mathcal{A}} u_a r(a) \\ \text{s.t.} \quad & \left(\sum_{t=1}^{T} q_t\right) \cdot \sum_{a\in\mathcal{A}} u_a c^{(i)}(a) \le B, \ \forall i \in [d] \\ & \sum_{a\in\mathcal{A}} u_a = 1 \\ & u_a \ge 0, \ \forall a \in \mathcal{A} \end{aligned} \tag{3}$$

The regret is similar:

$$\mathrm{Regret} = \mathrm{OPT} - \mathbb{E}\left[\sum_{t=1}^{\tau} q_t r_t\right]. \tag{4}$$

Algorithm

We design the following algorithm. At each round $t = 1, 2, ..., T$,

1. Receive q_t.
2. Construct estimator $\hat{Q}_t$ for $Q = \sum_{t=1}^{T} q_t$.
3. Construct confidence interval for $r(a)$, $c^{(i)}(a)$ for all $a \in \mathcal{A}$.
4. Solve "approximate" LP relaxation and get solution $\mathbf{p}_t$.
5. Select arm a with probability $\mathbf{p}_{t,a}$.
6. Receive reward and consume resource.
7. If some resource has been depleted, **Break**.

Regret Analysis

The Regret can be expressed by following:

$$\text{Regret} = \epsilon_1 + \epsilon_2 + \epsilon_3 + \epsilon_4 \tag{5}$$

where

- ϵ_1 represents the error between real outcome r_t and expected reward $r(A_t)$.
- ϵ_2 represents the error brings from confidence interval.
- ϵ_3 represents the error brings from random selection of A_t.
- ϵ_4 represents the estimation error of Q.

Still work in progress for lower and upper bound...

Summary

- Applications and examples for BwK.
- Generalized problem setting, benchmark, and regret for BwK.
- BwK with adversarial scaling: work in progress...

Thank you.

Shipra Agrawal and Nikhil R Devanur. Bandits with concave rewards and convex knapsacks. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 989–1006, 2014.

Shipra Agrawal, Nikhil R Devanur, and Lihong Li. An efficient algorithm for contextual bandits with knapsacks, and an extension to concave objectives. In *Conference on Learning Theory*, pages 4–18. PMLR, 2016.

Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 207–216. IEEE, 2013.

Nicole Immorlica, Karthik Abinav Sankararaman, Robert Schapire, and Aleksandrs Slivkins. Adversarial bandits with knapsacks. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 202–219. IEEE, 2019.

Karthik Abinav Sankararaman and Aleksandrs Slivkins. Combinatorial semi-bandits with knapsacks. In *International*

Conference on Artificial Intelligence and Statistics, pages 1760–1770. PMLR, 2018.