Algorithms and Uncertainty, Winter 2025/26 Lecture 25 (5 pages)

Bandits with Knapsacks

Thomas Kesselheim Last Update: January 27, 2026

We have seen different variants of the multi-armed bandit problem, particularly, the stochastic and the adversarial version with algorithms UCB1 and Exp3. All these problems have in common that the decision in one step does not affect what we can do in later steps. Today, we will get to know a very simple but general problem that introduces constraints between steps. For example, we can restrict how often an action is taken.

1 Bandits with Knapsacks

There are n actions, one of which is special and called the NULL action. There are T time steps. Furthermore, there are d resources, each with budget $B \geq 0$. In step $t = 1, \dots, T$, the algorithm chooses one action I_t. This raises reward $r_{I_t}^{(t)} \in [0,1]$ but also consumes $c_{I_t,j}^{(t)} \in [0,1]$ units of each resource j. For the NULL action, both the rewards and the resource consumptions are 0. As soon as one of the resources has been fully consumed, i.e., there is a j such that $\sum_{t'=1}^{t} c_{I_{t'},j}^{(t')} \geq B$, the algorithm can only choose the NULL action in the following steps. The goal is to maximize the sum of rewards $\sum_{t=1}^{T} r_{I_t}^{(t)}$.

We assume that rewards and resource consumptions are drawn from unknown probability distributions and they are revealed to us only after having chosen the action for the respective step. Within a time step, there is arbitrary correlation but random draws are independent across time steps. It will be convenient for us to also define rewards $r_i^{(t)}$ and resource consumptions $c_{i,j}^{(t)}$ for the actions i the algorithm does not choose in step t.

Example 25.1. *Suppose we have B units of an item. Sequentially, T buyers will arrive who are willing to pay different amounts for an item. In each step, the algorithm can choose a price at which one copy of the item will be offered to the current buyer. If the buyer accepts, the algorithm collects the price from the buyer as reward. Otherwise, it does not get any reward.*

This setting can be captured as Bandits with Knapsacks with one resource (i.e. $d = 1$). Let $v^{(t)}$ be the largest price the t-th buyer is willing to pay. The actions correspond to different prices. For every price p, we have

$$r_p^{(t)} = \begin{cases} p & \text{if } v^{(t)} \geq p \\ 0 & \text{otherwise} \end{cases} \qquad c_{p,1}^{(t)} = \begin{cases} 1 & \text{if } v^{(t)} \geq p \\ 0 & \text{otherwise} \end{cases}$$

Note that in our framework we have only finitely many actions and therefore also only finitely many prices. Furthermore, we have a NULL *action, which corresponds to not offering the item for sale at all.*

2 Bounding the Optimal Policy

In order to evaluate an algorithm in this setting, we will need to define a notion of regret. However, unlike in the settings we have studied so far, it does not make sense to compare to a single action. This is because the budget usually will not be high enough to allow one action to be taken throughout the entire sequence. But what would we do if we knew the probability distributions beforehand? Indeed, this is a non-trivial Markov decision process. Fortunately, we can bound the expected reward of any policy quite easily by a linear program as follows.

Lemma 25.2. *The expected reward of any policy (even if it knows the probability distributions) is upper-bounded by the optimal solution to the following LP with a variable x_i for every action i.*

$$\begin{aligned} &\textit{maximize } T \cdot \sum_{i=1}^{n} \bar{r}_i x_i \\ &\textit{subject to } \sum_{i=1}^{n} \bar{c}_{i,j} x_i \leq \frac{B+1}{T} && \textit{for all } j \\ &\qquad \sum_{i=1}^{n} x_i = 1 \\ &\qquad x_i \geq 0 && \textit{for all } i \end{aligned}$$

Here $\bar{r}_i = \mathbf{E}\left[r_i^{(t)}\right]$ and $\bar{c}_{i,j} = \mathbf{E}\left[c_{i,j}^{(t)}\right]$ are the expected rewards and consumptions of action i.

Proof. Consider any policy. Let $X_i^{(t)} = 1$ if the policy chooses action i in step t and 0 otherwise. We will show that $x_i = \frac{1}{T}\sum_{t=1}^{T} \mathbf{E}\left[X_i^{(t)}\right]$ corresponds to a solution of the LP.

Note that the policy chooses the action without knowing the reward and resource consumption in step t. Therefore, we have $\mathbf{E}\left[r_i^{(t)} X_i^{(t)}\right] = \mathbf{E}\left[r_i^{(t)}\right]\mathbf{E}\left[X_i^{(t)}\right] = \bar{r}_i \mathbf{E}\left[X_i^{(t)}\right]$. So the expected sum of rewards is

$$\mathbf{E}\left[\sum_{t=1}^{T} r_{I_t}^{(t)}\right] = \mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} r_i^{(t)} X_i^{(t)}\right] = \sum_{t=1}^{T}\sum_{i=1}^{n} \bar{r}_i \mathbf{E}\left[X_i^{(t)}\right] = T \cdot \sum_{i=1}^{n} \bar{r}_i x_i \ .$$

It only remains to verify that x indeed fulfills the constraints. For every resource j, we have $\sum_{t=1}^{T} c_{I_t,j}^{(t)} \leq B+1$ with probability 1, which means that also $\mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} c_{i,j}^{(t)} X_i^{(t)}\right] = \mathbf{E}\left[\sum_{t=1}^{T} c_{I_t,j}^{(t)}\right] \leq B+1$. We also have $\mathbf{E}\left[c_{i,j}^{(t)} X_i^{(t)}\right] = \mathbf{E}\left[c_{i,j}^{(t)}\right]\mathbf{E}\left[X_i^{(t)}\right] = \bar{c}_{i,j}\mathbf{E}\left[X_i^{(t)}\right]$. So

$$\sum_{i=1}^{n} \bar{c}_{i,j} x_i = \sum_{i=1}^{n} \bar{c}_{i,j} \frac{1}{T}\sum_{t=1}^{T} \mathbf{E}\left[X_i^{(t)}\right] = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n} \mathbf{E}\left[c_{i,j}^{(t)} X_i^{(t)}\right] = \frac{1}{T}\mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} c_{i,j}^{(t)} X_i^{(t)}\right] \leq \frac{B+1}{T} \ .$$

Finally, as the policy chooses exactly one action in every step, we have $\sum_{i=1}^{n} X_i^{(t)} = 1$ for all t, implying that $\sum_{i=1}^{n} x_i = \frac{1}{T}\sum_{t=1}^{T} \mathbf{E}\left[\sum_{i=1}^{n} X_i^{(t)}\right] = 1$. □

3 Algorithm

Our algorithm is a smart combination of two no-regret algorithms: We will use an algorithm for an adversarial bandits problem to choose the action. We could use an algorithm like Exp3[1]. This algorithm we also call the *primal* algorithm. We have to tell the primal algorithm how good the actions were. To this end, we define some surrogate reward $R_i^{(t)} \in \mathbb{R}$ for the i-th action in the t-th step. It should capture how much actual reward we obtain but also how much budget we consume.

[1]There is a technicality here that the algorithm needs to work against an adaptive adversary. This is an aspect we do not cover in class. It is enough to know that a slightly adapted version of Exp3 called Exp3.P works against an adaptive adversary.

How do we define $R_i^{(t)}$? First of all, we define a "fake resource" representing time. This resource has a budget of B and every action including the NULL action, consumes $\frac{B}{T}$ units of it per time step. This simplifies our life because eventually the algorithm will definitely run out of budget, possibly because the time horizon has been reached.

Letting τ denote the last step in which the algorithm is still within budget, we set

$$R_i^{(t)} = \begin{cases} r_i^{(t)} - \lambda \sum_{j=1}^{d+1} y_j^{(t)} c_{i,j}^{(t)} & \text{if } t \le \tau \\ 0 & \text{otherwise} \end{cases} ,$$

where $\lambda = \frac{T}{B}$ and $y_1^{(t)}, \dots, y_{d+1}^{(t)}$ still need to be defined. What's the intuition behind this expression? It is inspired by Lagrange functions but this is not too important. The idea is that we get some reward, represented by $r_i^{(t)}$ but we also spend some of our resources, time at the very least. The factors $y_1^{(t)}, \dots, y_{d+1}^{(t)} \ge 0$ add up to 1 and they give a weighting of how important each resource is.

Where do $y_1^{(t)}, \dots, y_{d+1}^{(t)}$ come from? They are chosen by another no-regret algorithm for an experts problem. We can use Multiplicative Weights. This algorithm is called the *dual* algorithm. The reward for choosing expert $j = 1, \dots, d+1$ is given by $c_{I_t,j}^{(t)}$ if $t \le \tau$ and 0 otherwise, so it depends on which action the primal algorithm chooses. (And the primal algorithm's reward in turn depends on what the dual algorithm chooses.)

We now define

$$\textsc{Regret}_{\text{primal}} = \max_i \sum_{t=1}^{T} R_i^{(t)} - \sum_{t=1}^{T} R_{I_t}^{(t)} \qquad \textsc{Regret}_{\text{dual}} = \max_j \sum_{t=1}^{\tau} c_{I_t,j}^{(t)} - \sum_{t=1}^{\tau} \sum_{j=1}^{d+1} y_j^{(t)} c_{I_t,j}^{(t)} .$$

Note that we can guarantee $\textsc{Regret}_{\text{primal}} = O(\lambda\sqrt{Tn}) = O(\frac{T}{B}\sqrt{T}n)$ and $\textsc{Regret}_{\text{dual}} = O(\sqrt{T \log d})$.

Theorem 25.3. *The expected reward of the Bandits with Knapsacks algorithm is at least*

$$\mathbf{E}[\text{ALG}] \ge \text{OPT} - \mathbf{E}[\textsc{Regret}_{\text{primal}}] - \frac{T}{B}\mathbf{E}[\textsc{Regret}_{\text{dual}}] - \frac{T}{B} ,$$

where OPT *is the value of an optimal solution to the LP. In particular, it is a no-regret algorithm if* $B = \Omega(T)$.

Proof. Recall that by introducing the resource representing time, the algorithm will definitely run out of budget for some resource, meaning that there is a resource j for which $\sum_{t=1}^{\tau} c_{I_t,j}^{(t)} \ge B$. By definition of the dual regret, we have

$$\sum_{t=1}^{\tau} \sum_{j=1}^{d+1} y_j^{(t)} c_{I_t,j}^{(t)} = \max_j \sum_{t=1}^{\tau} c_{I_t,j}^{(t)} - \textsc{Regret}_{\text{dual}} \ge B - \textsc{Regret}_{\text{dual}}$$

Let x^* be an optimal solution to the LP. As $\sum_{i=1}^{n} x_i^* = 1$, we have

$$\sum_{t=1}^{T} R_{I_t}^{(t)} \ge \sum_{i=1}^{n} \sum_{t=1}^{T} R_i^{(t)} x_i^* - \textsc{Regret}_{\text{primal}} .$$

In combination, we obtain

$$\begin{aligned} \text{ALG} &= \sum_{t=1}^{T} r_{I_t}^{(t)} = \sum_{t=1}^{T} R_{I_t}^{(t)} + \lambda \sum_{t=1}^{\tau} \sum_{j=1}^{d+1} y_j^{(t)} c_{I_t,j}^{(t)} \\ &\ge \sum_{t=1}^{T} \sum_{i=1}^{n} R_i^{(t)} x_i^* - \textsc{Regret}_{\text{primal}} + \lambda (B - \textsc{Regret}_{\text{dual}}) . \end{aligned}$$

Below, we will show

$$\mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} R_i^{(t)} x_i^*\right] \geq \frac{\mathbf{E}[\tau]}{T}\mathrm{OPT} - \mathbf{E}[\tau]\,\lambda\frac{B+1}{T} \quad . \tag{1}$$

As $\lambda = \frac{T}{B}$ and $\mathrm{OPT} \leq T$, this will then imply

$$\begin{aligned}
\mathbf{E}[\mathrm{ALG}] &\geq \mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} R_i^{(t)} x_i^*\right] + \lambda B - \mathbf{E}[\textsc{Regret}_{\mathrm{primal}}] - \lambda\mathbf{E}[\textsc{Regret}_{\mathrm{dual}}] \\
&\geq \frac{\mathbf{E}[\tau]}{T}\mathrm{OPT} - \mathbf{E}[\tau]\,\lambda\frac{B+1}{T} + \lambda B - \mathbf{E}[\textsc{Regret}_{\mathrm{primal}}] - \lambda\mathbf{E}[\textsc{Regret}_{\mathrm{dual}}] \\
&= \frac{\mathbf{E}[\tau]}{T}\mathrm{OPT} + \left(1 - \frac{\mathbf{E}[\tau]}{T}\right)T - \frac{\mathbf{E}[\tau]}{B} - \mathbf{E}[\textsc{Regret}_{\mathrm{primal}}] - \frac{T}{B}\mathbf{E}[\textsc{Regret}_{\mathrm{dual}}] \\
&\geq \mathrm{OPT} - \frac{T}{B} - \mathbf{E}[\textsc{Regret}_{\mathrm{primal}}] - \frac{T}{B}\mathbf{E}[\textsc{Regret}_{\mathrm{dual}}] \quad .
\end{aligned}$$

So, it only remains to show (1). To this end, let $Z_t = 1$ if $t \leq \tau$ and $Z_t = 0$ otherwise. We then have

$$\mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} R_i^{(t)} x_i^*\right] = \mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} Z_t\left(r_i^{(t)} - \lambda\sum_{j=1}^{d+1} y_j^{(t)} c_{i,j}^{(t)}\right) x_i^*\right]$$

By linearity of expectation

$$\mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} Z_t\left(r_i^{(t)} - \lambda\sum_{j=1}^{d+1} y_j^{(t)} c_{i,j}^{(t)}\right) x_i^*\right] = \sum_{t=1}^{T}\sum_{i=1}^{n}\mathbf{E}\left[Z_t r_i^{(t)}\right] x_i^* - \lambda\sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{d+1}\mathbf{E}\left[Z_t y_j^{(t)} c_{i,j}^{(t)}\right] x_i^* \quad .$$

Note that the rewards and consumptions in step t are independent of previous steps. Therefore $\mathbf{E}\left[Z_t r_i^{(t)}\right] = \mathbf{E}[Z_t]\,\mathbf{E}\left[r_i^{(t)}\right] = \mathbf{E}[Z_t]\,\bar{r}_i$ and $\mathbf{E}\left[Z_t y_j^{(t)} c_{i,j}^{(t)}\right] = \mathbf{E}\left[Z_t y_j^{(t)}\right]\mathbf{E}\left[c_{i,j}^{(t)}\right] = \mathbf{E}\left[Z_t y_j^{(t)}\right]\bar{c}_{i,j}^{(t)}$. Furthermore, we note that, as x^* is the optimal solution to the LP, we have $\sum_{i=1}^{n}\bar{r}_i x_i^* = \frac{\mathrm{OPT}}{T}$ and $\sum_{i=1}^{n}\bar{c}_{i,j}x_i^* \leq \frac{B+1}{T}$. So, as $\sum_{j=1}^{d+1} y_j^{(t)} = 1$ we also have $\sum_{i=1}^{n}\sum_{j=1}^{d+1} y_j^{(t)}\bar{c}_{i,j}x_i^* \leq \frac{B+1}{T}$. Consequently, we have

$$\begin{aligned}
\mathbf{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} R_i^{(t)} x_i^*\right] &= \mathbf{E}\left[\sum_{t=1}^{T} Z_t\left(\sum_{i=1}^{n}\bar{r}_i x_i^* - \lambda\sum_{i=1}^{n}\sum_{j=1}^{d+1} y_j^{(t)}\bar{c}_{i,j}x_i^*\right)\right] \\
&\geq \mathbf{E}\left[\sum_{t=1}^{T} Z_t\left(\frac{\mathrm{OPT}}{T} - \lambda\frac{B+1}{T}\right)\right] \\
&= \mathbf{E}[\tau]\left(\frac{\mathrm{OPT}}{T} - \lambda\frac{B+1}{T}\right) \quad . \qquad \square
\end{aligned}$$

4 Further Directions

There are a number of interesting research directions that have been investigated. For example, there is also an adversarial version of the Bandits with Knapsacks problem. However, there are no no-regret algorithms for it. Furthermore, one can improve the regret bounds. In particular, one can also obtain bounds for the case that B does not grow linearly in T.

5 Further Reading

- Ashwinkumar Badanidiyuru, Robert Kleinberg, Aleksandrs Slivkins: Bandits with Knapsacks. J. ACM 65(3): 13:1-13:55 (2018): First paper formalizing Bandits with Knapsacks.
- Nicole Immorlica, Karthik Abinav Sankararaman, Robert E. Schapire, Aleksandrs Slivkins: Adversarial Bandits with Knapsacks. J. ACM 69(6): 40:1-40:47 (2022): Paper introducing the algorithm.
- Thomas Kesselheim, Sahil Singla: Online Learning with Vector Costs and Bandits with Knapsacks. COLT 2020: 2286-2305: A generalization to ℓ_p norms.