

# Announcements

➤ HW 1 is due now

# CS650 I: Topics in Learning and Game Theory (Fall 2019)

## Adversarial Multi-Armed Bandits

---

Instructor: Haifeng Xu

# Outline

- The Adversarial Multi-armed Bandit Problem
- A Basic Algorithm: Exp3
- Regret Analysis of Exp3

# Recap: Online Learning So Far

Setup:  $T$  rounds; the following occurs at round  $t$ :

1. Learner picks a distribution  $p_t$  over actions  $[n]$
2. Adversary picks cost vector  $c_t \in [0,1]^n$
3. Action  $i_t \sim p_t$  is chosen and learner incurs cost  $c_t(i_t)$
4. Learner observes  $c_t$  (for use in future time steps)

Performance is typically measured by **regret**:

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

The multiplicative weight update algorithm has regret  $O(\sqrt{T \ln n})$ .

# Recap: Online Learning So Far

Convergence to equilibrium

- In repeated zero-sum games, if both players use a no-regret learning algorithm, their average strategy converges to an NE
- In general games, the average strategy converges to a CCE

Swap regret – a “stronger” regret concept and better convergence

- Def: each action  $i$  has a chance to deviate to another action  $s(i)$
- In repeated general games, if both players use a no-swap-regret learning algorithm, their average strategy converges to a CE

There is a general reduction, converting any learning algorithm with regret  $R$  to one with swap regret  $nR$ .

# This Lecture: Address Partial Feedback

- In online learning, the whole cost vector  $c_t$  can be observed by the learner, despite she only takes a single action  $i_t$ 
  - Realistic in some applications, e.g., stock investment
- In many cases, we only see the reward of the action we take
  - For example: slot machines, a.k.a., [multi-armed bandits](#)



# Other Applications with Partial Feedback

- Online advertisement placement or web ranking
  - Action: ad placement or ranking of webs
  - Cannot see the feedback for untaken actions

The screenshot shows a Google search for "pirate pants". The search bar is at the top with the Google logo and a search button. Below the search bar, there are tabs for "Web", "Shopping", "Images", "Videos", "News", "More", and "Search tools". The search results show "About 1,990,000 results (0.51 seconds)".

**Sponsored Ads:**

- Shop for pirate pants on Google** (Sponsored)
- Men Pirate Pants at Amazon** (Ads): [www.amazon.com/fashion](http://www.amazon.com/fashion), 4.4 ★★★★★ rating for amazon.com. Shop hundreds of favorite brands. Free Shipping on Qualified Orders.
- Pirate Print Pants**: [www.loudmouthgolf.com/Pants](http://www.loudmouthgolf.com/Pants). Fashion That Comes In Loud Colors. Choose Your Style. Order Now!
- Pirate Pants & Trousers**: [www.tobeapirate.com/](http://www.tobeapirate.com/). Complete your Pirate Outfit with authentic-design Pirate Pants.
- Target™ - Pirate Pants Kids**: [www.target.com/](http://www.target.com/), 4.3 ★★★★★ rating for target.com. Free Shipping On All Orders \$25+. Shop Pirate Pants Kids at Target™. 2099 Skokie Valley Rd, Highland Park (847) 266-8022.
- Pirate Pants 75% off**: [www.sale-fire.com/Pirate+Pants](http://www.sale-fire.com/Pirate+Pants). Save on Pirate Pants. Order today with free shipping! See your ad here »

**Search Results:**

- Renaissance Medieval Pirat...**: \$47.95, ToBeAPirate....
- Joma Sport Youth Combi...**: \$23.19, Epic Sports
- Velvet Pirate Adult Womens...**: \$16.99, TrendyHallow...
- Joma Sport Adult Combi P...**: \$23.19, Epic Sports
- Pirate Pants, Brown, XL 29...**: \$39.00, By The Sword

**Images for pirate pants** (Report images)

**More images for pirate pants**

**Dress Like A Pirate - Dresslikeapirate.com**: <https://dresslikeapirate.com/>. Wench Garb, Gypsy Jewels, Frock Coats, Velvet Vests, Pirate Shirts, Lace Jabots, Harem Pants, Pirate Boots, Bellydance Wear, Leather Belts, Bodices, Gypsy ... Dress Like a Pirate - Pirate Men - Pirate Wenches - All Women's

**Pirate Pants, Knee Breeches N Slops – Pirate Fashions**: [piratefashions.com/collections/pirate-pants-knee-breeches-n-slops](http://piratefashions.com/collections/pirate-pants-knee-breeches-n-slops). We have many options for ye: 2 versions of the classic Knee Breeches fer pirates who want confort, Buccaneer Pants fer gentlemen of fortune, n' 2 versions of th.

**Pirate clothing, nirate shirts, Pirate Pants, Pirate Boots, and**

# Other Applications with Partial Feedback

- Online advertisement placement or web ranking
  - Action: ad placement or ranking of webs
  - Cannot see the feedback for untaken actions
- Recommendation system:
  - Action = recommended option (e.g., a restaurant)
  - Do not know other options' feedback

The screenshot shows the Yelp website interface for Lexington, MA. At the top, there is a search bar with the text "Search for (e.g. taco, cheap dinner, Max's)" and a "Near (Address, City, State or Zip)" field containing "Lexington, MA 02420". The Yelp logo is on the left. Below the search bar is a navigation menu with links: "Welcome", "About Me", "Write a Review", "Find Reviews", "Find Friends", "Messaging", "Talk", "Events", and "Member Search".

The main content area is titled "Yelp Lexington" and includes a list of cities: "Boston", "San Francisco", "New York", "San Jose", "Los Angeles", "Chicago", and "More Cities". A prominent yellow banner reads "Yelp is the best way to find great local businesses" with a subtext: "People use Yelp to search for everything from the city's tastiest burger to the most renowned cardiologist. What will you uncover in your neighborhood?" and a "Create Your Free Account" button.

Below the banner is a section titled "The Best of Lexington" with a list of categories: "Restaurants" (5,575 reviewed), "Nightlife" (940 reviewed), "Food" (2,960 reviewed), "Shopping" (4,337 reviewed), "Bars" (684 reviewed), and "American (New)" (424 reviewed). The "Restaurants" category is expanded to show a list of top-rated businesses:

- 1. Royal India Bistro (61 reviews, Category: Indian). A review by David O. says: "I had my favorite chicken tikka masala and it was really..."
- 2. Wagon Wheel Nursery and Farm Stand (29 reviews)

On the right side of the page, there is a "Review of the Day" section featuring a review by Sarah D. for Beantown Taqueria (5 stars). The review text is: "I have been putting off writing this because I always get the same thing and I feel like I should probably branch out, but, nah. Beantown Carnitas tacos. Hot, if you're nasty. Medium if you're a lady.... Read more". Below this is an "Archive" link.

At the bottom right, there is a "Yelp on the Go" section with an image of a smartphone displaying the app and a "Get it for free now" button.

# Other Applications with Partial Feedback

- Online advertisement placement or web ranking
  - Action: ad placement or ranking of webs
  - Cannot see the feedback for untaken actions
- Recommendation system:
  - Action = recommended option (e.g., a restaurant)
  - Do not know other options' feedback
- Clinical trials
  - Action = a treatment
  - Don't know what would happen for treatments not chosen
- Playing strategic games
  - Cannot observe opponents' strategies but only know the payoff of the taken action
  - E.g., Poker games, competition in markets

# Adversarial Multi-Armed Bandits (MAB)

- Very much like online learning, except **partial feedback**
  - The name “bandit” is inspired by slot machines
- Model: at each time step  $t = 1, \dots, T$ ; the following occurs in order
  1. Learner picks a distribution  $p_t$  over **arms**  $[n]$
  2. Adversary picks cost vector  $c_t \in [0,1]^n$
  3. **Arm**  $i_t \sim p_t$  is chosen and learner incurs cost  $c_t(i_t)$
  4. Learner **only observes**  $c_t(i_t)$  (for use in future time steps)
- Though we cannot observe  $c_t$ , adversary still picks  $c_t$  **before**  $i_t$  is sampled

Q: since learner does not observe  $c_t(i)$  for  $i \neq i_t$ , can adversary arbitrarily modify these  $c_t(i)$ 's after  $i_t$  has been selected?

No, because this makes  $c_t$  depends on sampled  $i_t$  which is not allowed

# Outline

- The Adversarial Multi-armed Bandit Problem
- A Basic Algorithm: Exp3
- Regret Analysis of Exp3

Recall the algorithm for full information setting:

Parameter:  $\epsilon$

Initialize weight  $w_1(i) = 1, \forall i = 1, \dots, n$

For  $t = 1, \dots, T$

1. Let  $W_t = \sum_{i \in [n]} w_t(i)$ , pick arm  $i$  with probability  $w_t(i)/W_t$
2. Observe cost vector  $c_t \in [0,1]^n$
3. For all  $i \in [n]$ , update  $w_{t+1}(i) = w_t(i) \cdot (1 - \epsilon c_t(i))$

Recall the algorithm for full information setting:

Parameter:  $\epsilon$

Initialize weight  $w_1(i) = 1, \forall i = 1, \dots, n$

For  $t = 1, \dots, T$

1. Let  $W_t = \sum_{i \in [n]} w_t(i)$ , pick arm  $i$  with probability  $w_t(i)/W_t$
2. Observe cost vector  $c_t \in [0,1]^n$
3. For all  $i \in [n]$ , update  $w_{t+1}(i) = w_t(i) \cdot e^{-\epsilon \cdot c_t(i)}$

Recall  $1 - \delta \approx e^{-\delta}$  for small  $\delta$

Recall the algorithm for full information setting:

Parameter:  $\epsilon$

Initialize weight  $w_1(i) = 1, \forall i = 1, \dots, n$

For  $t = 1, \dots, T$

1. Let  $W_t = \sum_{i \in [n]} w_t(i)$ , pick arm  $i$  with probability  $w_t(i)/W_t$
2. Observe cost vector  $c_t \in [0,1]^n$
3. For all  $i \in [n]$ , update  $w_{t+1}(i) = w_t(i) \cdot e^{-\epsilon \cdot c_t(i)}$

- In this lecture we will use this **exponential-weight** variant, and prove its regret bound en route
- Also called *Exponential Weight Update (EWU)*

Recall  $1 - \delta \approx e^{-\delta}$  for small  $\delta$

Recall the algorithm for full information setting:

Parameter:  $\epsilon$

Initialize weight  $w_1(i) = 1, \forall i = 1, \dots, n$

For  $t = 1, \dots, T$

1. Let  $W_t = \sum_{i \in [n]} w_t(i)$ , pick arm  $i$  with probability  $w_t(i)/W_t$
2. Observe cost vector  $c_t \in [0,1]^n$
3. For all  $i \in [n]$ , update  $w_{t+1}(i) = w_t(i) \cdot e^{-\epsilon \cdot c_t(i)}$

Basic idea of Exp3

- Want to use EWU, but do not know vector  $c_t \rightarrow$  try to estimate  $c_t$ !
- Well, we really only have  $c_t(i_t)$ , what can we do?

Estimate  $\bar{c}_t = (0, \dots, 0, c_t(i_t), 0, \dots, 0)^T$ ?  Too optimistic

Estimate  $\bar{c}_t = \left(0, \dots, 0, \frac{c_t(i_t)}{p_t(i_t)}, 0, \dots, 0\right)^T$  

# Exp3: a Basic Algorithm for Adversarial MAB

Parameter:  $\epsilon$

Initialize weight  $w_1(i) = 1, \forall i = 1, \dots, n$

For  $t = 1, \dots, T$

1. Let  $W_t = \sum_{i \in [n]} w_t(i)$ , pick arm  $i$  with probability  $w_t(i)/W_t$
2. Observe cost vector  $c_t \in [0,1]^n$
3. For all  $i \in [n]$ , update  $w_{t+1}(i) = w_t(i) \cdot e^{-\epsilon \cdot \bar{c}_t(i)}$  where  $\bar{c}_t = (0, \dots, 0, c_t(i_t)/p_t(i_t), 0, \dots, 0)^T$ .

- That is, weight is updated only for the pulled arm
  - Because we really don't know how good are other arms at  $t$
  - But  $i_t$  is more heavily penalized now
  - Attention:  $c_t(i_t)/p_t(i_t)$  may be extremely large if  $p_t(i_t)$  is small
- Called **Exp3**: **Ex**ponential-weight algorithm for **Exp**loration and **Exp**loitation

# A Closer Look at the Estimator $\bar{c}_t$

- $\bar{c}_t$  is random – it depends on the randomly sampled  $i_t \sim p_t$
- $\bar{c}_t$  is an unbiased estimator of  $c_t$ , i.e.,  $\mathbb{E}_{i_t \sim p_t} \bar{c}_t = c_t$ 
  - Because given  $p_t$ , for any  $i$  we have

$$\begin{aligned}\mathbb{E}_{i_t \sim p_t} \bar{c}_t(i) &= \mathbb{P}(i_t = i) \cdot \frac{c_t(i)}{p_t(i)} + \mathbb{P}(i_t \neq i) \cdot 0 \\ &= p_t(i) \cdot \frac{c_t(i)}{p_t(i)} \\ &= c_t(i)\end{aligned}$$

- This is exactly the reason for our choice of  $\bar{c}_t$

# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

## Some key differences from online learning

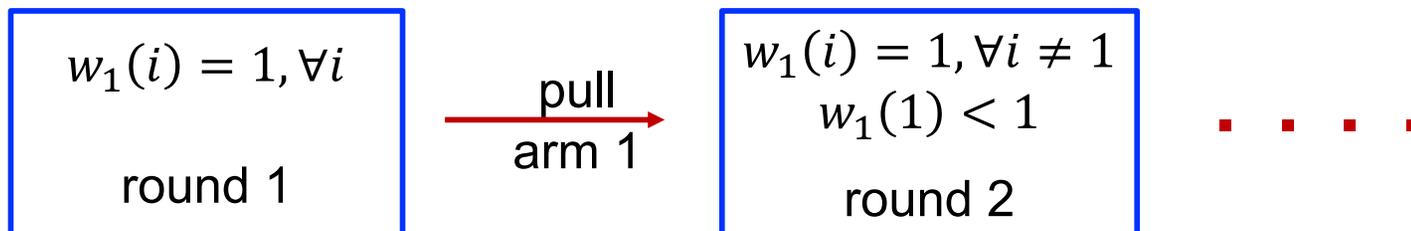
- $R_T$  is random (even it already takes expectation over  $i_t \sim p_t$ )
  - Because distribution  $p_t$  itself is random, depends on sampled  $i_1, \dots, i_{t-1}$
  - That is, if we run the same algorithm for multiple times, we will get different  $R_T$  value even when facing the same adversary!

# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

## Some key differences from online learning

- $R_T$  is random (even it already takes expectation over  $i_t \sim p_t$ )
- Because distribution  $p_t$  itself is random, depends on sampled  $i_1, \dots, i_{t-1}$
  - That is, if we run the same algorithm for multiple times, we will get different  $R_T$  value even when facing the same adversary!

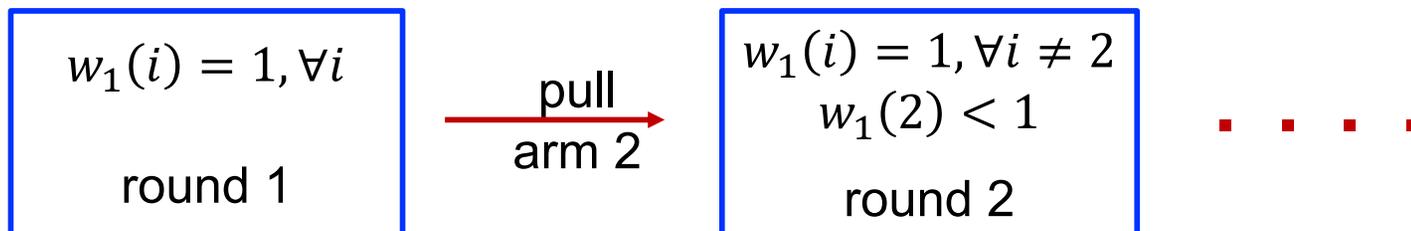


# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

## Some key differences from online learning

- $R_T$  is random (even it already takes expectation over  $i_t \sim p_t$ )
- Because distribution  $p_t$  itself is random, depends on sampled  $i_1, \dots, i_{t-1}$
  - That is, if we run the same algorithm for multiple times, we will get different  $R_T$  value even when facing the same adversary!



# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

## Some key differences from online learning

- $R_T$  is random (even it already takes expectation over  $i_t \sim p_t$ )
  - Because distribution  $p_t$  itself is random, depends on sampled  $i_1, \dots, i_{t-1}$
  - That is, if we run the same algorithm for multiple times, we will get different  $R_T$  value even when facing the same adversary
- Cost vector  $c_t$  is also random as it generally depends on  $p_t$ 
  - Adversary maps distribution  $p_t$  to a cost vector  $c_t$
- This is not the case in online learning
  - If we run the same algorithm for multiple times, we shall obtain the same  $R_T$  value if facing the same adversary

# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

- Therefore, in principle, we have to upper bound  $\mathbb{E}(R_T)$  where expectation is over the randomness of arm sampling

$$\begin{aligned} \mathbb{E}(R_T) &= \mathbb{E} \left[ \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right] \\ &= \sum_{i \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(i) p_t(i)] - \mathbb{E} \left[ \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right] \end{aligned}$$

by linearity of expectation

# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

- Therefore, in principle, we have to upper bound  $\mathbb{E}(R_T)$  where expectation is over the randomness of arm sampling

$$\begin{aligned} \mathbb{E}(R_T) &= \mathbb{E} \left[ \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right] \\ &= \sum_{i \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(i) p_t(i)] - \mathbb{E} \left[ \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right] \\ &\geq \sum_{i \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(i) p_t(i)] - \min_{j \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(j)] \end{aligned}$$

because  $\min_{j \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(j)] \geq \mathbb{E} \left[ \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right]$

(proof: homework exercise)

# Regret

$$R_T = \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j)$$

- Therefore, in principle, we have to upper bound  $\mathbb{E}(R_T)$  where expectation is over the randomness of arm sampling

$$\begin{aligned} \mathbb{E}(R_T) &= \mathbb{E} \left[ \sum_{i \in [n]} \sum_{t \in [T]} c_t(i) p_t(i) - \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right] \\ &= \sum_{i \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(i) p_t(i)] - \mathbb{E} \left[ \min_{j \in [n]} \sum_{t \in [T]} c_t(j) \right] \\ &\geq \underbrace{\sum_{i \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(i) p_t(i)] - \min_{j \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(j)]}_{\text{Pseudo-Regret } \overline{R_T}} \end{aligned}$$

- Good regret guarantees good pseudo-regret, but not the reverse

## Bounding regret turns out to be challenging

- Exp3 is not sufficient to guarantee small regret
- Next, we instead prove that Exp3 has small **pseudo-regret**
  - As is typical in many works
- A slight modification of Exp3 can be proved to have small regret

# Outline

- The Adversarial Multi-armed Bandit Problem
- A Basic Algorithm: Exp3
- Regret Analysis of Exp3

**Theorem.** The pseudo regret of Exp3 is  $O(\sqrt{nT \ln n})$ .

High-level idea of the proof

- Pretend to be in the full information setting with cost equal the estimated  $\bar{c}_t$
- Relate  $\bar{c}_t$  to  $c_t$  since we know it is an unbiased estimator of  $c_t$

# Imitate a Full-Info Setting with Cost $\bar{c}_t$

- Recall regret bound for full information setting

$$R_T^{full} \leq \frac{\ln n}{\epsilon} + \epsilon T$$

- This holds for any cost vector, thus also  $\bar{c}_t$
- But...one issue is that  $\bar{c}_t(i_t)$  may be greater than 1
- Not a big issue – the same analysis yields the following bound

$$R_T^{full} \leq \frac{\ln n}{\epsilon} + \epsilon \max_i \sum_{t \in [T]} [\bar{c}_t(i)]^2$$

Real Issue:  $\bar{c}_t(i)$  may be too large that we cannot bound  $R_T^{full}$

# Imitate a Full-Info Setting with Cost $\bar{c}_t$

A regret bound as follows turns out to work for our proof

$$R_T^{full} \leq \frac{\ln n}{\epsilon} + \epsilon \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$$

- That is, instead of  $\max_i$ , the bound here averages over  $i$
- Why more useful?
  - The  $p_t(i)$  term will help to cancel out a  $p_t(i)$  denominator in  $\bar{c}_t(i) = c_t(i)/p_t(i)$
  - This turns out to be enough to bound the regret

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

Parameter:  $\epsilon$

Initialize weight  $w_1(i) = 1, \forall i = 1, \dots, n$

For  $t = 1, \dots, T$

1. Let  $W_t = \sum_{i \in [n]} w_t(i)$ , pick arm  $i$  with probability  $w_t(i)/W_t$
2. Observe cost vector  $\bar{c}_t \geq 0$
3. For all  $i \in [n]$ , update  $w_{t+1}(i) = w_t(i) \cdot e^{-\epsilon \cdot \bar{c}_t(i)}$

Note: this yields a bound  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} T$  when  $c_t \in [0,1]^n$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

Proof: similar technique – carefully bound certain quantity

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

Proof: similar technique – carefully bound certain quantity

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

Why this term?

- It tracks weight decrease (will be clear in next slide)
- The algebraic reasons,  $e^{-\delta} \approx 1 - \delta + \delta^2/2$ , which will give rise to both the term  $p_t(i) \bar{c}_t(i)$  and  $p_t(i) [\bar{c}_t(i)]^2$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

**Fact 1.**  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)} = W_{t+1} / W_t$ , where  $W_t = \sum_i w_t(i)$ .

- The term  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$  is the decreasing rate of  $W_t$
- Formal proof: HW exercise

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

**Fact 1.**  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)} = W_{t+1} / W_t$ , where  $W_t = \sum_i w_t(i)$ .

- The term  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$  is the decreasing rate of  $W_t$
- Formal proof: HW exercise

**Corollary.**  $\sum_t \log \left[ \sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)} \right] = \log W_{T+1} - \log n$

- Telescope sum and  $W_1 = n$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

**Fact 2.**  $\sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] \leq -\epsilon \sum_{t,i} p_t(i) c_t(i) + \frac{\epsilon^2}{2} \sum_{t,i} p_t(i) [c_t(i)]^2$ .

Follows from algebraic calculation

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

**Fact 2.**  $\sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] \leq -\epsilon \sum_{t,i} p_t(i) c_t(i) + \frac{\epsilon^2}{2} \sum_{t,i} p_t(i) [c_t(i)]^2$ .

Follows from algebraic calculation

$$\sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] \leq \sum_t \log\left[\sum_{i \in [n]} p_t(i) \left[1 - \epsilon c_t(i) + \frac{\epsilon^2}{2} [c_t(i)]^2\right]\right]$$

By  $e^{-\delta} \leq 1 - \delta + \delta^2/2$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

**Fact 2.**  $\sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] \leq -\epsilon \sum_{t,i} p_t(i) c_t(i) + \frac{\epsilon^2}{2} \sum_{t,i} p_t(i) [c_t(i)]^2$ .

Follows from algebraic calculation

$$\begin{aligned} \sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] &\leq \sum_t \log\left[\sum_{i \in [n]} p_t(i) \left[1 - \epsilon c_t(i) + \frac{\epsilon^2}{2} [c_t(i)]^2\right]\right] \\ &= \sum_t \log\left[1 - \sum_{i \in [n]} p_t(i) \epsilon c_t(i) + \sum_{i \in [n]} p_t(i) \frac{\epsilon^2}{2} [c_t(i)]^2\right] \end{aligned}$$

Since  $\sum_{i \in [n]} p_t(i) = 1$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

➤ Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$

**Fact 2.**  $\sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] \leq -\epsilon \sum_{t,i} p_t(i) c_t(i) + \frac{\epsilon^2}{2} \sum_{t,i} p_t(i) [c_t(i)]^2$ .

Follows from algebraic calculation

$$\begin{aligned} \sum_t \log[\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}] &\leq \sum_t \log\left[\sum_{i \in [n]} p_t(i) \left[1 - \epsilon c_t(i) + \frac{\epsilon^2}{2} [c_t(i)]^2\right]\right] \\ &= \sum_t \log\left[1 - \sum_{i \in [n]} p_t(i) \epsilon c_t(i) + \sum_{i \in [n]} p_t(i) \frac{\epsilon^2}{2} [c_t(i)]^2\right] \\ &\leq -\epsilon \sum_{t,i} p_t(i) c_t(i) + \frac{\epsilon^2}{2} \sum_{t,i} p_t(i) [c_t(i)]^2 \end{aligned}$$

Since  $\log(1 + \delta) \leq \delta$  for any  $\delta$

# Step 1: Tighter Regret for Full-Info Case

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

- Consider quantity  $\sum_{i \in [n]} p_t(i) e^{-\epsilon c_t(i)}$
- Combining the two facts yields the lemma
  - HW exercise

## Step 2: Relate $\bar{c}_t$ to Pseudo-Regret

Recall pseudo-regret definition

$$\begin{aligned}\bar{R}_T &= \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t] - \min_{j \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(j)] \\ &= \max_{j \in [n]} \left[ \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t] - \sum_{t \in [T]} \mathbb{E}[c_t(j)] \right] \\ &= \max_{j \in [n]} \underbrace{\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)]}_{\text{Pseudo-regret from action } j}\end{aligned}$$

## Step 2: Relate $\bar{c}_t$ to Pseudo-Regret

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

- That is, expected pseudo regret from  $j$  w.r.t. true cost  $c_t$  equals that w.r.t. the estimated cost  $\bar{c}_t$

Recall pseudo-regret definition

$$\begin{aligned} \bar{R}_T &= \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t] - \min_{j \in [n]} \sum_{t \in [T]} \mathbb{E}[c_t(j)] \\ &= \max_{j \in [n]} \left[ \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t] - \sum_{t \in [T]} \mathbb{E}[c_t(j)] \right] \\ &= \max_{j \in [n]} \underbrace{\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)]}_{\text{Pseudo-regret from action } j} \end{aligned}$$

## Step 2: Relate $\bar{c}_t$ to Pseudo-Regret

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ Proof:

$$\mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)] = \mathbb{E}[\mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j) | p_t]]$$

Because the randomness of  $\bar{c}_t$  comes:

1. Randomness of  $i_t \sim p_t$
2. Randomness of  $p_t$  itself which depends on  $i_1, \dots, i_{t-1}$

## Step 2: Relate $\bar{c}_t$ to Pseudo-Regret

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ Proof:

$$\begin{aligned} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)] &= \mathbb{E}[\mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j) | p_t]] \\ &= \mathbb{E}[\mathbb{E}[c_t \cdot p_t - c_t(j) | p_t]] \end{aligned}$$

Because conditioning on  $p_t$ ,  $\bar{c}_t$  is an unbiased estimator of  $c_t$

## Step 2: Relate $\bar{c}_t$ to Pseudo-Regret

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ Proof:

$$\begin{aligned} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)] &= \mathbb{E}[\mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j) | p_t]] \\ &= \mathbb{E}[\mathbb{E}[c_t \cdot p_t - c_t(j) | p_t]] \\ &= \mathbb{E}[c_t \cdot p_t - c_t(j)] \end{aligned}$$

## Step 3: Derive Pseudo-Regret Bounds

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ For any  $j$ , we have

$$\begin{aligned} \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] &= \mathbb{E}\left[\sum_{t \in [T]} [\bar{c}_t \cdot p_t - \bar{c}_t(j)]\right] \\ &\leq \mathbb{E}\left[\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2\right] \end{aligned}$$

By Lemma 1

## Step 3: Derive Pseudo-Regret Bounds

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ For any  $j$ , we have

$$\begin{aligned} \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] &= \mathbb{E}[\sum_{t \in [T]} [\bar{c}_t \cdot p_t - \bar{c}_t(j)]] \\ &\leq \mathbb{E} \left[ \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 \right] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} \left[ \mathbb{E}[\sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 | p_t] \right] \end{aligned}$$

By conditional expectation

## Step 3: Derive Pseudo-Regret Bounds

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ For any  $j$ , we have

$$\begin{aligned} \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] &= \mathbb{E}[\sum_{t \in [T]} [\bar{c}_t \cdot p_t - \bar{c}_t(j)]] \\ &\leq \mathbb{E} \left[ \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 \right] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [ \mathbb{E}[\sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 | p_t] ] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [\sum_t \sum_i p_t(i) \mathbb{E}[[\bar{c}_t(i)]^2 | p_t]] \end{aligned}$$

By linearity of expectation

## Step 3: Derive Pseudo-Regret Bounds

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ For any  $j$ , we have

$$\begin{aligned} \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] &= \mathbb{E}[\sum_{t \in [T]} [\bar{c}_t \cdot p_t - \bar{c}_t(j)]] \\ &\leq \mathbb{E} \left[ \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 \right] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [ \mathbb{E}[\sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 | p_t] ] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [\sum_t \sum_i p_t(i) \mathbb{E}[[\bar{c}_t(i)]^2 | p_t]] \end{aligned}$$

$$\text{Observer } \mathbb{E}[[\bar{c}_t(i)]^2 | p_t] = 0 \cdot [1 - p_t(i)] + \left[ \frac{c_t(i)}{p_t(i)} \right]^2 \cdot p_t(i) = \frac{[c_t(i)]^2}{p_t(i)}$$

## Step 3: Derive Pseudo-Regret Bounds

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ For any  $j$ , we have

$$\begin{aligned} \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] &= \mathbb{E}[\sum_{t \in [T]} [\bar{c}_t \cdot p_t - \bar{c}_t(j)]] \\ &\leq \mathbb{E} \left[ \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 \right] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [ \mathbb{E}[\sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 | p_t] ] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [\sum_t \sum_i \boxed{p_t(i)} \mathbb{E}[[\bar{c}_t(i)]^2 | p_t]] \end{aligned}$$

$$\text{Observer } \mathbb{E}[[\bar{c}_t(i)]^2 | p_t] = 0 \cdot [1 - p_t(i)] + \left[ \frac{c_t(i)}{p_t(i)} \right]^2 \cdot p_t(i) = \boxed{\frac{[c_t(i)]^2}{p_t(i)}}$$

## Step 3: Derive Pseudo-Regret Bounds

**Lemma 1.** The regret of the following algorithm is at most  $\frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2$  for any cost vector  $\bar{c}_t \geq 0$ .

**Lemma 2.**  $\sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] = \sum_{t \in [T]} \mathbb{E}[\bar{c}_t \cdot p_t - \bar{c}_t(j)]$

➤ For any  $j$ , we have

$$\begin{aligned} \sum_{t \in [T]} \mathbb{E}[c_t \cdot p_t - c_t(j)] &= \mathbb{E}[\sum_{t \in [T]} [\bar{c}_t \cdot p_t - \bar{c}_t(j)]] \\ &\leq \mathbb{E} \left[ \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 \right] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [ \mathbb{E}[\sum_t \sum_i p_t(i) [\bar{c}_t(i)]^2 | p_t] ] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E} [\sum_t \sum_i p_t(i) \mathbb{E}[[\bar{c}_t(i)]^2 | p_t]] \\ &= \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} \mathbb{E}[\sum_t \sum_i [c_t(i)]^2] \\ &\leq \frac{\ln n}{\epsilon} + \frac{\epsilon}{2} nT \end{aligned}$$

Pick  $\epsilon = \sqrt{\frac{2 \ln n}{nT}}$  yields a regret bound of  $O(\sqrt{nT \ln n})$

# Summary of the Proof

- A tighter regret bound for full information setting
- Treat the (realized) estimated  $\bar{c}_t$  as the cost for full information
- Expected pseudo-regret w.r.t. to  $c_t$  equals expected pseudo-regret w.r.t. to  $\bar{c}_t$
- Upper bound pseudo-regret by taking expectation over  $\bar{c}_t$ 's

# The True Regret and Beyond

- Exp3 does not guarantee good true regret, still because  $c_t(i)/p_t(i)$  may be too large
  - Pseudo-regret “smooths out”  $p_t(i)$  by taking expectations first
- To obtain good true regret, need to modify Exp3 by adding some uniform exploration so that  $p_t(i)$  is never too small
  - More intricate analysis, but will get the same regret bound  $O(\sqrt{nT \ln n})$
- In addition to adversarial feedback, a “nicer” setting is when the cost of each arm is drawn from a **fixed but unknown** distribution
  - Called stochastic multi-armed bandits
  - Naturally, Exp3 and regret bound  $O(\sqrt{nT \ln n})$  still applies
  - But a better algorithm called Upper-Confidence Bounds (UCB) yields much better regret bound  $O(\sqrt{n \ln T})$
  - Different analysis techniques

# Thank You

Haifeng Xu

University of Virginia

[hx4ad@virginia.edu](mailto:hx4ad@virginia.edu)

