RLearning:
Short guides to reinforcement learning

Unit 1-1: Multi-Armed Bandits

Davud Rostam-Afschar (Uni Mannheim)

How to assign treatments adaptively?

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow “earning while learning”
- Push to replace non-adaptive randomized trials with bandits

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow “earning while learning”
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, . . .
  - Practitioners use bandit algorithms

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow "earning while learning"
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, ...
  - Practitioners use bandit algorithms
  - Can improve outcomes for participants (optimize regret)
  - Can improve policies learned at the end of trial (best-arm identification)

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow "earning while learning"
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, ...
  - Practitioners use bandit algorithms
  - Can improve outcomes for participants (optimize regret)
  - Can improve policies learned at the end of trial (best-arm identification)
- Some popular algorithms

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow "earning while learning"
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, . . .
  - Practitioners use bandit algorithms
  - Can improve outcomes for participants (optimize regret)
  - Can improve policies learned at the end of trial (best-arm identification)
- Some popular algorithms
  - ε-first

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow "earning while learning"
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, ...
  - Practitioners use bandit algorithms
  - Can improve outcomes for participants (optimize regret)
  - Can improve policies learned at the end of trial (best-arm identification)
- Some popular algorithms
  - ε-first
  - ε-greedy

Adaptive Experimental Designs

- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow "earning while learning"
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, ...
  - Practitioners use bandit algorithms
  - Can improve outcomes for participants (optimize regret)
  - Can improve policies learned at the end of trial (best-arm identification)
- Some popular algorithms
  - ε-first
  - ε-greedy
  - Upper Confidence Bound

Adaptive Experimental Designs

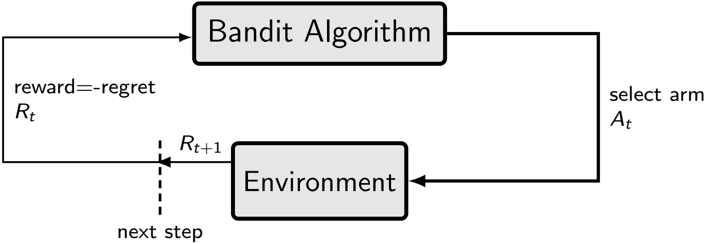
- Randomized controlled trials gold standard of causal inference
- Adaptive experiments allow "earning while learning"
- Push to replace non-adaptive randomized trials with bandits
  - In medicine, economics, political science, survey methods research, education, psychology, ...
  - Practitioners use bandit algorithms
  - Can improve outcomes for participants (optimize regret)
  - Can improve policies learned at the end of trial (best-arm identification)
- Some popular algorithms
  - ε-first
  - ε-greedy
  - Upper Confidence Bound
  - Thompson sampling

Stylized Data Structure

Obs	Selected Arm	Reward
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

- Does arm A or arm B perform better?
- Which arm to play in next trial (round 17)?

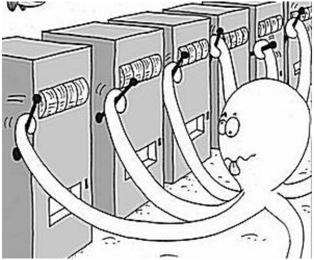
Multi-Armed Bandits as a Reinforcement Learning Problem



Goal: Learn to choose actions that maximize rewards

What Are Multi-Armed Bandits?

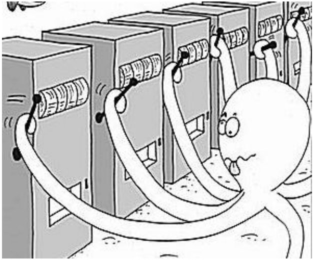
- A sequential decision-making problem
- Agent chooses among K options ("arms") repeatedly
- Each arm gives an unknown reward
- Objective: Maximize total reward (or minimize **regret**)
- Core tradeoff: Learning vs. Earning



Microsoft Research

What Are Multi-Armed Bandits?

- A sequential decision-making problem
- Agent chooses among K options ("arms") repeatedly
- Each arm gives an unknown reward
- Objective: Maximize total reward (or minimize **regret**)
- Core tradeoff: Learning vs. Earning



Microsoft Research

But how to balance **exploration** and **exploitation**?

Bandit Algorithms

The Exploration/Exploitation Dilemma

- The action-value is the true but unknown mean reward for action a:

$$q(a) = \mathbb{E}[R_t \mid A_t = a], \quad \forall a \in \{1, \ldots, k\}$$

- Estimate expected return:

$$Q_t(a) \approx q(a), \quad \forall a \text{ (action-value estimates)}$$

- Define the greedy action at time t as:

$$A_t^* = \arg\max_a Q_t(a)$$

- If $A_t = A_t^*$ then you are **exploiting**
- If $A_t \neq A_t^*$ then you are **exploring**

Regret

- The optimal value is:

$$r^* = q(a^*) = \max_{a \in \mathcal{A}} q(a)$$

- The regret is the opportunity loss for one step:

$$\text{loss}_t = \mathbb{E}[r^* - q(a_t)]$$

- The total regret is the total opportunity loss:

$$\text{Loss}_T = \sum_{t=1}^{T} \text{loss}_t = \mathbb{E}\left[\sum_{t=1}^{T} r^* - q(a_t)\right]$$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

1. **Compute counts & empirical means**
   - Total pulls: $T = 16$
   - Arm A: $N_{16}(A) = 10$, $\sum_{t:a_t=A} r_t = 6$

$$\implies q(A) = 6/10 = 0.6$$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

1. **Compute counts & empirical means**
   - Total pulls: $T = 16$
   - Arm A: $N_{16}(A) = 10$, $\sum_{t:a_t=A} r_t = 6$

     $\implies q(A) = 6/10 = 0.6$

   - Arm B: $N_{16}(B) = 6$, $\sum_{t:a_t=B} r_t = 1$

     $\implies q(B) = 1/6 \approx 0.1667$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

1. **Compute counts & empirical means**
   - Total pulls: $T = 16$
   - Arm A: $N_{16}(A) = 10$, $\sum_{t:a_t=A} r_t = 6$

     $$\implies q(A) = 6/10 = 0.6$$

   - Arm B: $N_{16}(B) = 6$, $\sum_{t:a_t=B} r_t = 1$

     $$\implies q(B) = 1/6 \approx 0.1667$$

2. **Identify the optimal arm and its mean**

$$r^* = \max\{q(A), q(B)\} = 0.6$$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

3. **Sum of per-round regrets**

$$\text{Loss}_{16} = \sum_{t=1}^{16} (r^* - q(a_t))$$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

3. **Sum of per-round regrets**

$$\text{Loss}_{16} = \sum_{t=1}^{16} (r^* - q(a_t))$$

- For $a_t = A$: $r^* - q(A) = 0.6 - 0.6 = 0$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

3. **Sum of per-round regrets**

$$\text{Loss}_{16} = \sum_{t=1}^{16} (r^* - q(a_t))$$

- For $a_t = A$: $r^* - q(A) = 0.6 - 0.6 = 0$
- For $a_t = B$: $r^* - q(B) = 0.6 - 0.1667 \approx 0.4333$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

3. **Sum of per-round regrets**

$$\text{Loss}_{16} = \sum_{t=1}^{16} (r^* - q(a_t))$$

- For $a_t = A$: $r^* - q(A) = 0.6 - 0.6 = 0$
- For $a_t = B$:
  $r^* - q(B) = 0.6 - 0.1667 \approx 0.4333$

  Played 6 times:

$$\text{Loss}_{16} = 6 \times 0.4333 \approx 2.6$$

Stylized Data Structure: Per-Arm = Per-Round Regrets

Obs t	Selected Arm a_t	Reward r_t
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

4. **Sum of per-arm regrets**

$$\sum_{a \in \{A,B\}} N_{16}(a)(r^* - q(a))$$

$$= 10 \times 0 + 6 \times 0.4333 \approx 2.6$$

Regret Decomposition Lemma

- Number of times action a has been selected prior to time t
  $N_t(a) = \sum_{i=1}^{t-1} \mathbb{1}\{A_i = a\}$
- Total regret can be rewritten as:

$$\text{Loss}_t = \mathbb{E}\left[\sum_{t=1}^{T} r^* - q(a_t)\right]$$

$$= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)](r^* - q(a))$$

- Regret comes from pulling suboptimal arms
- Each arm a contributes $(r^* - q(a))$ for every time $N_t(a)$ it's chosen

Stylized Data Structure

t	Arm	Reward
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

Stylized Data Structure

t	Arm	Reward
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

Stylized Data Structure

t	Arm	Reward	$\sum_{i=1}^{n} R_i$
1	A	0	0
2	B	0	
3	A	1	1
4	B	0	
5	A	0	1
6	B	1	
7	A	1	2
8	B	0	
9	A	0	2
10	A	1	3
11	A	1	4
12	B	0	
13	A	1	5
14	A	0	5
15	A	1	6
16	B	0	

Stylized Data Structure

t	Arm	Reward	$\sum_{i=1}^{n} R_i$	n
1	A	0	0	1
2	B	0		
3	A	1	1	2
4	B	0		
5	A	0	1	3
6	B	1		
7	A	1	2	4
8	B	0		
9	A	0	2	5
10	A	1	3	6
11	A	1	4	7
12	B	0		
13	A	1	5	8
14	A	0	5	9
15	A	1	6	10
16	B	0		

Stylized Data Structure

t	Arm	Reward	$\sum_{i=1}^{n} R_i$	n	Q_n
1	A	0	0	1	0.00
2	B	0			
3	A	1	1	2	0.50
4	B	0			
5	A	0	1	3	0.33
6	B	1			
7	A	1	2	4	0.50
8	B	0			
9	A	0	2	5	0.40
10	A	1	3	6	0.50
11	A	1	4	7	0.57
12	B	0			
13	A	1	5	8	0.63
14	A	0	5	9	0.56
15	A	1	6	10	0.60
16	B	0			

Stylized Data Structure

t	Arm	Reward	$\sum_{i=1}^{n} R_i$	n	Q_n	Q_{n-1}
1	A	0	0	1	0.00	0.00
2	B	0				
3	A	1	1	2	0.50	0.00
4	B	0				
5	A	0	1	3	0.33	0.50
6	B	1				
7	A	1	2	4	0.50	0.33
8	B	0				
9	A	0	2	5	0.40	0.50
10	A	1	3	6	0.50	0.40
11	A	1	4	7	0.57	0.50
12	B	0				
13	A	1	5	8	0.63	0.57
14	A	0	5	9	0.56	0.63
15	A	1	6	10	0.60	0.56
16	B	0				

Incremental Update

- Let $R_1, \ldots, R_n$ be rewards received after selecting an action n times.
- Define Q_n as the estimate after $n-1$ rewards:

$$Q_n = \frac{1}{n-1} \sum_{i=1}^{n-1} R_i$$

- Now, after receiving the n-th reward R_n, we update:

$$Q_{n+1} = \frac{1}{n} \sum_{i=1}^{n} R_i = \frac{1}{n} \left(R_n + \sum_{i=1}^{n-1} R_i \right)$$

$$= \frac{1}{n} \left(R_n + (n-1) Q_n \right) = Q_n + \frac{1}{n} \left(R_n - Q_n \right)$$

Incremental Update

- Let $R_1, \ldots, R_n$ be rewards received after selecting an action n times.
- Define Q_n as the estimate after $n-1$ rewards:

$$Q_n = \frac{1}{n-1} \sum_{i=1}^{n-1} R_i$$

- Now, after receiving the n-th reward R_n, we update:

$$Q_{n+1} = \frac{1}{n} \sum_{i=1}^{n} R_i = \frac{1}{n} \left(R_n + \sum_{i=1}^{n-1} R_i \right)$$

$$= \frac{1}{n} (R_n + (n-1) Q_n) = Q_n + \frac{1}{n} (R_n - Q_n)$$

- This is the incremental update rule:

$$\underbrace{Q_{n+1}}_{\text{New}} = \underbrace{Q_n}_{\text{Old}} + \underbrace{\frac{1}{n}}_{\text{Step size}} \underbrace{(R_n - Q_n)}_{\text{Error}}$$

Incremental Update

- Let $R_1, \ldots, R_n$ be rewards received after selecting an action n times.
- Define Q_n as the estimate after $n-1$ rewards:

$$Q_n = \frac{1}{n-1} \sum_{i=1}^{n-1} R_i$$

- Now, after receiving the n-th reward R_n, we update:

$$Q_{n+1} = \frac{1}{n} \sum_{i=1}^{n} R_i = \frac{1}{n} \left(R_n + \sum_{i=1}^{n-1} R_i \right)$$
$$= \frac{1}{n} (R_n + (n-1) Q_n) = Q_n + \frac{1}{n} (R_n - Q_n)$$

- This is the incremental update rule:

$$\underbrace{Q_{n+1}}_{\text{New}} = \underbrace{Q_n}_{\text{Old}} + \underbrace{\frac{1}{n}}_{\text{Step size}} \underbrace{(R_n - Q_n)}_{\text{Error}}$$

We can get a new estimate without storing all the rewards.

Stylized Data Structure: Learning Expected Return

t	Arm	Reward
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

Stylized Data Structure: Learning Expected Return

t	Arm	Reward
1	A	0
2	B	0
3	A	1
4	B	0
5	A	0
6	B	1
7	A	1
8	B	0
9	A	0
10	A	1
11	A	1
12	B	0
13	A	1
14	A	0
15	A	1
16	B	0

Stylized Data Structure: Learning Expected Return

t	Arm	Reward	Q_{n-1}
1	A	0	0.00
2	B	0	
3	A	1	0.00
4	B	0	
5	A	0	0.50
6	B	1	
7	A	1	0.33
8	B	0	
9	A	0	0.50
10	A	1	0.40
11	A	1	0.50
12	B	0	
13	A	1	0.57
14	A	0	0.63
15	A	1	0.56
16	B	0	

Stylized Data Structure: Learning Expected Return

t	Arm	Reward	Q_{n-1}	Update
1	A	0	0.00	$0.00 + \frac{1}{1}(0 - 0.00)$
2	B	0		
3	A	1	0.00	$0.00 + \frac{1}{2}(1 - 0.00)$
4	B	0		
5	A	0	0.50	$0.50 + \frac{1}{3}(0 - 0.50)$
6	B	1		
7	A	1	0.33	$0.33 + \frac{1}{4}(1 - 0.33)$
8	B	0		
9	A	0	0.50	$0.50 + \frac{1}{5}(0 - 0.50)$
10	A	1	0.40	$0.40 + \frac{1}{6}(1 - 0.40)$
11	A	1	0.50	$0.50 + \frac{1}{7}(1 - 0.50)$
12	B	0		
13	A	1	0.57	$0.57 + \frac{1}{8}(1 - 0.57)$
14	A	0	0.63	$0.63 + \frac{1}{9}(0 - 0.63)$
15	A	1	0.56	$0.56 + \frac{1}{10}(1 - 0.56)$
16	B	0		

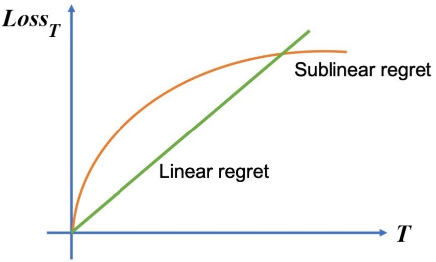
Stylized Data Structure: Learning Expected Return

t	Arm	Reward	Q_{n-1}	Update	Q_n
1	A	0	0.00	$0.00 + \frac{1}{1}(0 - 0.00)$	0.00
2	B	0			
3	A	1	0.00	$0.00 + \frac{1}{2}(1 - 0.00)$	0.50
4	B	0			
5	A	0	0.50	$0.50 + \frac{1}{3}(0 - 0.50)$	0.33
6	B	1			
7	A	1	0.33	$0.33 + \frac{1}{4}(1 - 0.33)$	0.50
8	B	0			
9	A	0	0.50	$0.50 + \frac{1}{5}(0 - 0.50)$	0.40
10	A	1	0.40	$0.40 + \frac{1}{6}(1 - 0.40)$	0.50
11	A	1	0.50	$0.50 + \frac{1}{7}(1 - 0.50)$	0.57
12	B	0			
13	A	1	0.57	$0.57 + \frac{1}{8}(1 - 0.57)$	0.63
14	A	0	0.63	$0.63 + \frac{1}{9}(0 - 0.63)$	0.56
15	A	1	0.56	$0.56 + \frac{1}{10}(1 - 0.56)$	0.60
16	B	0			

Sublinear Regret

Most multi-armed bandit (MAB) algorithms aim to achieve sublinear regret, so that the *average* regret vanishes as the number of rounds $T \to \infty$:

$$\lim_{T \to \infty} \frac{\text{Loss}_T}{T} = 0$$



- Loss_T: expected total expected regret after T rounds.

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$
ε-first	$\mathcal{O}(T)$

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$
ε-first	$\mathcal{O}(T)$
ε-greedy	$\mathcal{O}(T)$

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$
ε-first	$\mathcal{O}(T)$
ε-greedy	$\mathcal{O}(T)$
ε-greedy (decaying)	$\mathcal{O}(\log T)$

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$
ε-first	$\mathcal{O}(T)$
ε-greedy	$\mathcal{O}(T)$
ε-greedy (decaying)	$\mathcal{O}(\log T)$
UCB (Upper Confidence Bound)	$\mathcal{O}(\log T)$

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$
ε*-first*	$\mathcal{O}(T)$
ε*-greedy*	$\mathcal{O}(T)$
ε*-greedy (decaying)*	$\mathcal{O}(\log T)$
UCB (Upper Confidence Bound)	$\mathcal{O}(\log T)$
Thompson Sampling	$\mathcal{O}(\log T)$

Exploration Strategies for Stochastic Bandits

Algorithm	**Total Regret**
greedy	$\mathcal{O}(T)$
ε-first	$\mathcal{O}(T)$
ε-greedy	$\mathcal{O}(T)$
ε-greedy (decaying)	$\mathcal{O}(\log T)$
UCB (Upper Confidence Bound)	$\mathcal{O}(\log T)$
Thompson Sampling	$\mathcal{O}(\log T)$

Overviews: **?**, **?**, **?**

Bandits in Practice

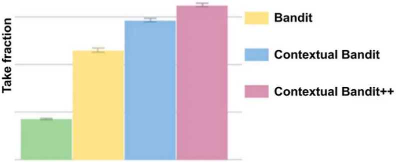
Real World Bandit: *Netflix Artwork*

- For a particular movie, which image to show to users on Netflix?
- **Actions:** Choose one of k images to display
- **Ground-truth mean rewards (unknown):**
  True percentage of users who click on image and watch movie
- **Estimated mean rewards:**
  Average observed click-through rates for each image

Profile Type	Score Image A	Score Image B
Comedy	5.7	6.3
Romance	7.2	6.5

Image A

Image B



Source: Netflix Tech Blog

References I

BUBECK, S., AND N. CESA-BIANCHI (2012): “Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems,” *Foundations and Trends® in Machine Learning*, 5(1), 1–122.

BURTINI, G., J. LOEPPKY, AND R. LAWRENCE (2015): “A survey of online experiment design with the stochastic multi-armed bandit,” *arXiv preprint arXiv:1510.00757*.

SLIVKINS, A. (2019): “Introduction to Multi-Armed Bandits,” *Foundations and Trends® in Machine Learning*, 12(1-2), 1–286.

Takeaways

How to Balance Earning and Learning?

- Multi-Armed Bandits (MAB) model adaptive, sequential decision-making under uncertainty
- Core trade-off: Exploration vs. Exploitation
- Objective: Maximize total reward, or equivalently, minimize regret
- Key algorithms:
  - ε-first, ε-greedy (fixed or decaying)
  - UCB (Upper Confidence Bound)
  - Thompson Sampling (Bayesian approach)

Appendix

Regret Decomposition Lemma

- Sample mean reward $\mathbb{E}[q(a_t)]$ is true mean reward $q(a)$ times average number of times actions a was chosen

$$\mathbb{E}[q(a_t)] = \mathbb{E}\Big[\sum_{a \in \mathcal{A}} q(a)\mathbb{1}\{A_i = a\}\Big] = \sum_{a \in \mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]$$

Regret Decomposition Lemma

- Sample mean reward $\mathbb{E}[q(a_t)]$ is true mean reward $q(a)$ times average number of times actions a was chosen

$$\mathbb{E}[q(a_t)] = \mathbb{E}\Big[\sum_{a \in \mathcal{A}} q(a)\mathbb{1}\{A_i = a\}\Big] = \sum_{a \in \mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]$$

- Total regret can be rewritten as:

$$\text{Loss}_t = \sum_{i=1}^{t-1} \mathbb{E}[r^* - q(a_t)]$$

Regret Decomposition Lemma

- Sample mean reward $\mathbb{E}[q(a_t)]$ is true mean reward $q(a)$ times average number of times actions a was chosen

$$\mathbb{E}[q(a_t)] = \mathbb{E}\left[\sum_{a \in \mathcal{A}} q(a)\mathbb{1}\{A_i = a\}\right] = \sum_{a \in \mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]$$

- Total regret can be rewritten as:

$$\text{Loss}_t = \sum_{i=1}^{t-1} \mathbb{E}[r^* - q(a_t)]$$

$$= \sum_{i=1}^{t-1} \left[\sum_{a \in \mathcal{A}} r^* \mathbb{E}[\mathbb{1}\{A_i = a\}] - \sum_{a \in \mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]\right]$$

Regret Decomposition Lemma

- Sample mean reward $\mathbb{E}[q(a_t)]$ is true mean reward $q(a)$ times average number of times actions a was chosen

$$\mathbb{E}[q(a_t)] = \mathbb{E}\left[\sum_{a\in\mathcal{A}} q(a)\mathbb{1}\{A_i = a\}\right] = \sum_{a\in\mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]$$

- Total regret can be rewritten as:

$$\text{Loss}_t = \sum_{i=1}^{t-1} \mathbb{E}[r^* - q(a_t)]$$

$$= \sum_{i=1}^{t-1}\left[\sum_{a\in\mathcal{A}} r^*\mathbb{E}[\mathbb{1}\{A_i = a\}] - \sum_{a\in\mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]\right]$$

$$= \sum_{i=1}^{t-1}\sum_{a\in\mathcal{A}} \mathbb{E}\left[\mathbb{1}\{A_i = a\}\right](r^* - q(a)) = \sum_{a\in\mathcal{A}} \mathbb{E}\left[\sum_{i=1}^{t-1} \mathbb{1}\{A_i = a\}\right](r^* - q(a)$$

Regret Decomposition Lemma

- Sample mean reward $\mathbb{E}[q(a_t)]$ is true mean reward $q(a)$ times average number of times actions a was chosen

$$\mathbb{E}[q(a_t)] = \mathbb{E}\left[\sum_{a\in\mathcal{A}} q(a)\mathbb{1}\{A_i = a\}\right] = \sum_{a\in\mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]$$

- Total regret can be rewritten as:

$$\text{Loss}_t = \sum_{i=1}^{t-1} \mathbb{E}[r^* - q(a_t)]$$

$$= \sum_{i=1}^{t-1} \left[\sum_{a\in\mathcal{A}} r^*\mathbb{E}[\mathbb{1}\{A_i = a\}] - \sum_{a\in\mathcal{A}} q(a)\mathbb{E}[\mathbb{1}\{A_i = a\}]\right]$$

$$= \sum_{i=1}^{t-1} \sum_{a\in\mathcal{A}} \mathbb{E}\left[\mathbb{1}\{A_i = a\}\right](r^* - q(a)) = \sum_{a\in\mathcal{A}} \mathbb{E}\left[\sum_{i=1}^{t-1} \mathbb{1}\{A_i = a\}\right](r^* - q(a)$$

$$= \sum_{a\in\mathcal{A}} \mathbb{E}[N_t(a)](r^* - q(a))$$

Exploration Strategies for Stochastic Bandits

Algorithm	Idea	Total Regret	Type
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist

Exploration Strategies for Stochastic Bandits

Algorithm	Idea	Total Regret	Type
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist
ε-first	Explore randomly for $\varepsilon \cdot T$ rounds, then exploit the best action	$\mathcal{O}(T)$	Frequentist

Exploration Strategies for Stochastic Bandits

Algorithm	Idea	Total Regret	Type
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist
ε-*first*	Explore randomly for $\varepsilon \cdot T$ rounds, then exploit the best action	$\mathcal{O}(T)$	Frequentist
ε-*greedy*	Pick the best action, except with probability ε choose randomly	$\mathcal{O}(T)$	Frequentist

Exploration Strategies for Stochastic Bandits

Algorithm	Idea	Total Regret	Type
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist
ε-first	Explore randomly for $\varepsilon \cdot T$ rounds, then exploit the best action	$\mathcal{O}(T)$	Frequentist
ε-greedy	Pick the best action, except with probability ε choose randomly	$\mathcal{O}(T)$	Frequentist
ε-greedy (decaying)	Same as above, but with a decaying ε over time	$\mathcal{O}(\log T)$	Frequentist

Exploration Strategies for Stochastic Bandits

Algorithm	**Idea**	**Total Regret**	**Type**
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist
ε-first	Explore randomly for $\varepsilon \cdot T$ rounds, then exploit the best action	$\mathcal{O}(T)$	Frequentist
ε-greedy	Pick the best action, except with probability ε choose randomly	$\mathcal{O}(T)$	Frequentist
ε-greedy (decaying)	Same as above, but with a decaying ε over time	$\mathcal{O}(\log T)$	Frequentist
UCB (Upper Confidence Bound)	Choose action with highest estimated mean plus uncertainty bonus	$\mathcal{O}(\log T)$	Frequentist

Exploration Strategies for Stochastic Bandits

Algorithm	**Idea**	**Total Regret**	**Type**
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist
ε-first	Explore randomly for $\varepsilon \cdot T$ rounds, then exploit the best action	$\mathcal{O}(T)$	Frequentist
ε-greedy	Pick the best action, except with probability ε choose randomly	$\mathcal{O}(T)$	Frequentist
ε-greedy (decaying)	Same as above, but with a decaying ε over time	$\mathcal{O}(\log T)$	Frequentist
UCB (Upper Confidence Bound)	Choose action with highest estimated mean plus uncertainty bonus	$\mathcal{O}(\log T)$	Frequentist
Thompson Sampling	Sample from posterior distribution of rewards and choose best action	$\mathcal{O}(\log T)$	Bayesian

Exploration Strategies for Stochastic Bandits

Algorithm	Idea	Total Regret	Type
greedy	Pick the best action in round t	$\mathcal{O}(T)$	Frequentist
ε-first	Explore randomly for $\varepsilon \cdot T$ rounds, then exploit the best action	$\mathcal{O}(T)$	Frequentist
ε-greedy	Pick the best action, except with probability ε choose randomly	$\mathcal{O}(T)$	Frequentist
ε-greedy (decaying)	Same as above, but with a decaying ε over time	$\mathcal{O}(\log T)$	Frequentist
UCB (Upper Confidence Bound)	Choose action with highest estimated mean plus uncertainty bonus	$\mathcal{O}(\log T)$	Frequentist
Thompson Sampling	Sample from posterior distribution of rewards and choose best action	$\mathcal{O}(\log T)$	Bayesian

Overviews: **?**, **?**, **?**