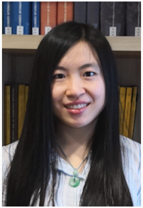
Pure Exploration in Multi-Armed Bandits

Zixin Zhong (University of Alberta)

Vincent Y. F. Tan (National University of Singapore)



National University of Singapore
Tutorial 2 in IJCAI 2022
25 July 2022

Outline

1. **What is multi-armed bandits (MAB)?**
   - Classification of MAB problems
   - Example — Cascading bandits
2. **Explore state-of-the-art findings of pure exploration**
   - BAI: fixed-confidence setting
   - BAI: fixed-budget setting
3. **Summary and discussions**

Outline

1. What is multi-armed bandits (MAB)?
2. Explore state-of-the-art findings of pure exploration
3. Summary and discussions

Motivation: data-driven optimization

NUS National University of Singapore

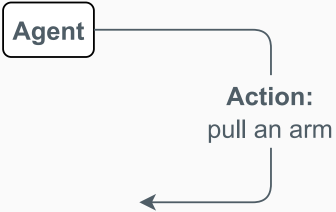
- Subdomain of reinforcement learning, online learning problem.
- Application:
    - Internet advertisement placement
    - Restaurant recommendation
    - Clinical trials
    - ......

YAHOO! yelp

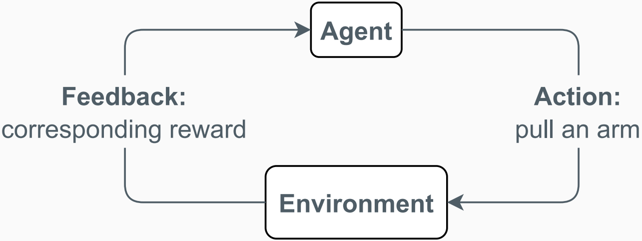
Style of tutorial:

- Will present a few well-known models/algorithms
- Will present some "newer" models/algorithms
- Since it's a tutorial, we will go through some proofs

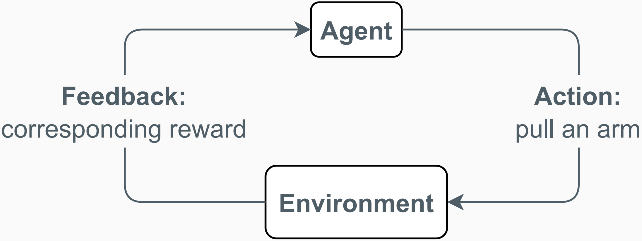
Multi-armed bandit problem (MAB)



Multi-armed bandit problem (MAB)

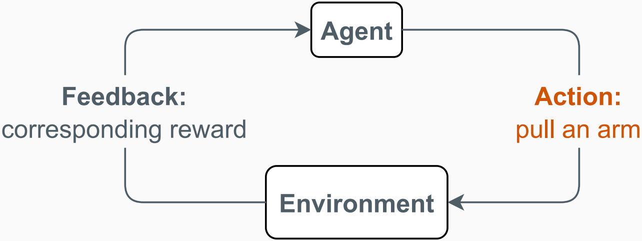


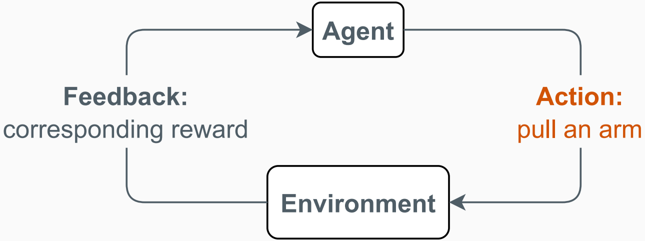
Multi-armed bandit problem (MAB)



Objectives

1. Maximize the cumulative reward over a fixed horizon.
2. Find the best arm (largest expected reward).

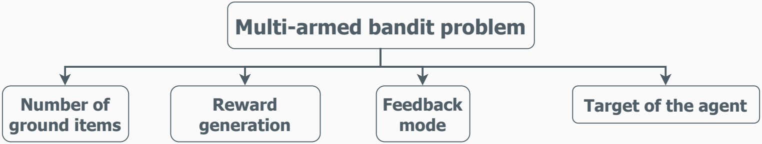

Agent
Feedback:
corresponding reward
Action:
pull an arm
Environment



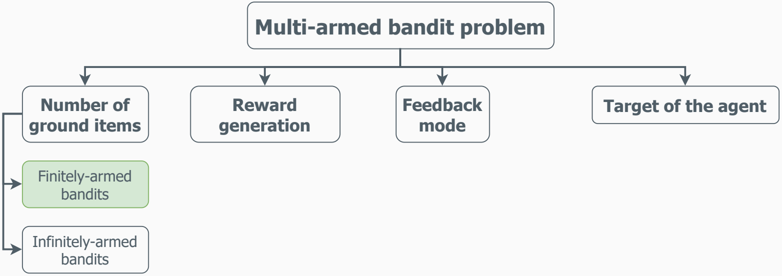
Challenge

- Exploitation: to pull "confident" arms to maximize reward.
- Exploration: to pull "unconfident" arms to find better ones.

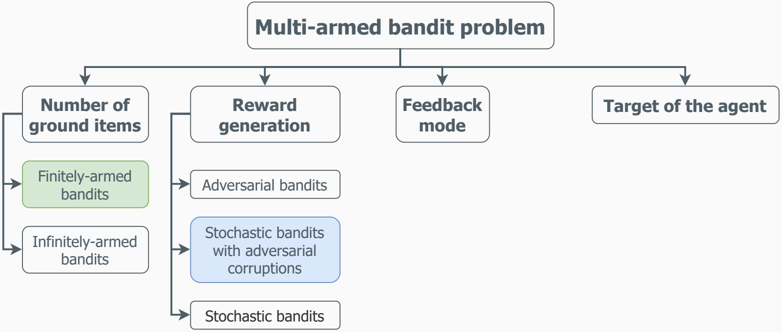
Classification



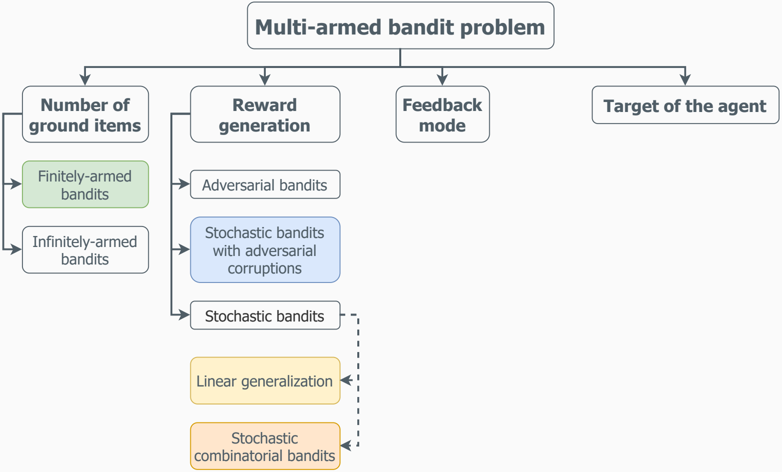
Classification

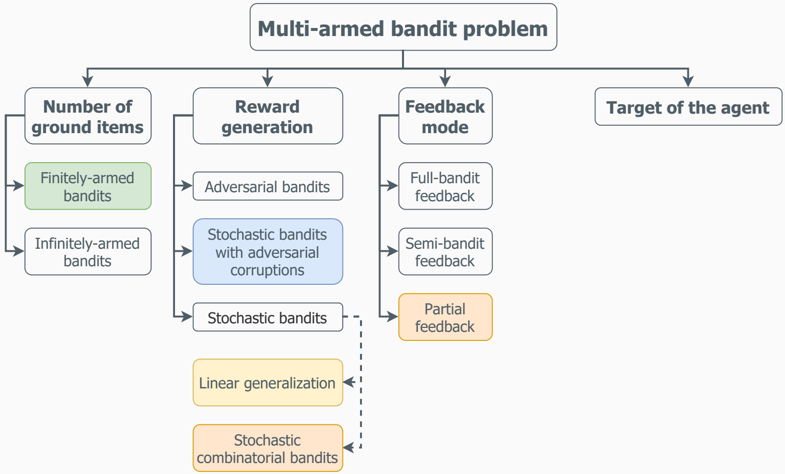


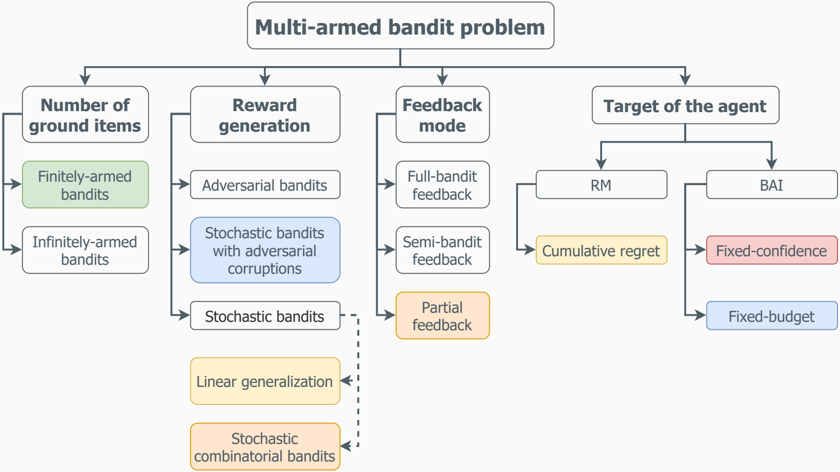
Classification



Classification




Multi-armed bandit problem
Number of ground items
Finitely-armed bandits
Infinitely-armed bandits
Reward generation
Adversarial bandits
Stochastic bandits with adversarial corruptions
Stochastic bandits
Linear generalization
Stochastic combinatorial bandits
Feedback mode
Full-bandit feedback
Semi-bandit feedback
Partial feedback
Target of the agent


Multi-armed bandit problem
Number of ground items
Finitely-armed bandits
Infinitely-armed bandits
Reward generation
Adversarial bandits
Stochastic bandits with adversarial corruptions
Stochastic bandits
Linear generalization
Stochastic combinatorial bandits
Feedback mode
Full-bandit feedback
Semi-bandit feedback
Partial feedback
Target of the agent
RM
Cumulative regret
BAI
Fixed-confidence
Fixed-budget

Formulation of MAB models

♠ **Ground set** — $\mathcal{S}$ consists of available arms.

♠ **Dynamics** — At each time step $t = 1, 2, \ldots$

1. **Reward** $W_t(i)$ is associated with arm i.
2. Agent **pulls** arm A_t
3. Agent observes the corresponding **feedback** $O_t = f(\{W_t(i) : i \in A_t\})$.

Formulation of MAB models

♠ **Ground set** — $\mathcal{S}$ consists of available arms.

♠ **Dynamics** — At each time step $t = 1, 2, \ldots$

1. **Reward** $W_t(i)$ is associated with arm i.
2. Agent **pulls** arm A_t
3. Agent observes the corresponding **feedback** $O_t = f(\{W_t(i) : i \in A_t\})$.

♠ **Number of arms**

- Finite-armed bandits (Audibert et al., 2009; Agrawal and Goyal, 2012)
  Ground set $\mathcal{S}$ of L arms is indexed by $[L] = \{1, 2, \ldots, L\}$.
- Infinite-armed bandits (Berry et al., 1997)
  Related to the topic of Bayesian optimization

Stochastic bandits

- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.

Stochastic bandits

- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.

♠ Linear generalization (Abe and Long, 1999)

- $w(i) = x(i)^\top \beta$
- Feature vector $x(i) \in \mathbb{R}^d$ is known for each arm i, latent vector $\beta \in \mathbb{R}^d$ is not known.
- Reduces to standard bandits when $x(i) = e_i$, standard basis.

Stochastic bandits

- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.

♠ Linear generalization (Abe and Long, 1999)

- $w(i) = x(i)^\top \beta$
- Feature vector $x(i) \in \mathbb{R}^d$ is known for each arm i, latent vector $\beta \in \mathbb{R}^d$ is not known.
- Reduces to standard bandits when $x(i) = e_i$, standard basis.

♠ Stochastic combinatorial bandits

- Stardard setting: $|A_t| = 1$.
- Combinatorial setting: $|A_t| \geq 1$.

Feedback mode

- ♠ Full-bandit feedback

- ♠ Semi-bandit feedback

- ♠ Partial feedback

Feedback mode

♠ Full-bandit feedback

Agent only observes the **sums** of the realizations of all pulled arms (Rejwan and Mansour, 2020; Kuroki et al., 2020).

♠ Semi-bandit feedback

♠ Partial feedback

♠ FULL-BANDIT FEEDBACK

Agent only observes the sums of the realizations of all pulled arms (Rejwan and Mansour, 2020; Kuroki et al., 2020).

♠ SEMI-BANDIT FEEDBACK

Agent observes realizations of all pulled arms (Mannor and Tsitsiklis, 2004; Kalyanakrishnan et al., 2012).

♠ PARTIAL FEEDBACK

Feedback mode

♠ **Full-bandit feedback**

Agent only observes the sums of the realizations of all pulled arms (Rejwan and Mansour, 2020; Kuroki et al., 2020).

♠ **Semi-bandit feedback**

Agent observes realizations of all pulled arms (Mannor and Tsitsiklis, 2004; Kalyanakrishnan et al., 2012).

♠ **Partial feedback**

Agent only observes the realizations of a subset of pulled arms (Kveton et al., 2015b; Li et al., 2016).

Reward generation

♠ Stochastic bandits

♠ STOCHASTIC BANDITS

♠ STOCHASTIC BANDITS WITH ADVERSARIAL CORRUPTIONS

(Shen, 2019; Jun et al., 2018)

At each time step $t = 1, \ldots, T$:

1. Stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each arm i.

♠ STOCHASTIC BANDITS

♠ STOCHASTIC BANDITS WITH ADVERSARIAL CORRUPTIONS

(Shen, 2019; Jun et al., 2018)

At each time step $t = 1, \dots, T$:

1. Stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each arm i.
2. Agent pulls arm i_t.
3. Adversary observes $\{W_t(i)\}_{i \in [L]}$ as well as i_t, and corrupts $W_t(i_t)$ with c_t:
$$\tilde{W}_t(i_t) = W_t(i_t) + c_t \in [0, 1].$$
   but the norm of $\{c_t\}_{t=1}^T$ is suitably constrained.
4. Agent observes the corrupted reward $\tilde{W}_t(i_t)$.

Reward generation

♠ Stochastic bandits

♠ Stochastic bandits with adversarial corruptions

(Shen, 2019; Jun et al., 2018)

At each time step $t = 1, \ldots, T$:

1. Stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each arm i.
2. Agent pulls arm i_t.
3. Adversary observes $\{W_t(i)\}_{i \in [L]}$ as well as i_t, and corrupts $W_t(i_t)$ with c_t:
$$\tilde{W}_t(i_t) = W_t(i_t) + c_t \in [0, 1].$$
but the norm of $\{c_t\}_{t=1}^{T}$ is suitably constrained.
4. Agent observes the corrupted reward $\tilde{W}_t(i_t)$.

♠ Stochastic bandits

♠ Stochastic bandits with adversarial corruptions

(Shen, 2019; Jun et al., 2018)

At each time step $t = 1, \ldots, T$:

1. Stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each arm i.
2. Agent pulls arm i_t.
3. Adversary observes $\{W_t(i)\}_{i \in [L]}$ as well as i_t, and corrupts $W_t(i_t)$ with c_t:
$$\tilde{W}_t(i_t) = W_t(i_t) + c_t \in [0, 1].$$
but the norm of $\{c_t\}_{t=1}^{T}$ is suitably constrained.
4. Agent observes the corrupted reward $\tilde{W}_t(i_t)$.

♠ Adversarial/Non-stochastic bandits

(Auer et al., 2002b; Cesa-Bianchi and Lugosi, 2006)

- Rewards $\{W_t(i)\}_{t=1}$ of each arm i are not necessarily drawn independently from the same distribution.

♠ Stochastic bandits

♠ Stochastic bandits with adversarial corruptions

(Shen, 2019; Jun et al., 2018)

At each time step $t = 1, \ldots, T$:

1. Stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each arm i.
2. Agent pulls arm i_t.
3. Adversary observes $\{W_t(i)\}_{i \in [L]}$ as well as i_t, and corrupts $W_t(i_t)$ with c_t:
$$\tilde{W}_t(i_t) = W_t(i_t) + c_t \in [0, 1].$$
but the norm of $\{c_t\}_{t=1}^T$ is suitably constrained.
4. Agent observes the corrupted reward $\tilde{W}_t(i_t)$.

♠ Adversarial/Non-stochastic bandits

(Auer et al., 2002b; Cesa-Bianchi and Lugosi, 2006)

- Rewards $\{W_t(i)\}_{t=1}$ of each arm i are not necessarily drawn independently from the same distribution.

Stochastically constrained adversarial bandits (Zimmert and Seldin, 2021)

- $W_t(i)$ is a r.v. with mean $w_t(i)$, and gaps $\Delta_{i,j} = W_t(i) - W_t(j)$ are fixed.

Target of the agent

♠ Cumulative regret minimization

♠ Simple regret minimization

♠ Pure exploration/best arm identification (BAI)

Fixed-confidence setting

Fixed-budget setting

Target of the agent

♠ Cumulative regret minimization

Maximize the cumulative reward, i.e.,minimize the regret (the gap between the maximum cumulative reward and the reward obtained by the agent) (Agrawal and Goyal, 2012; Russo and Van Roy, 2014; Lai, 1987).

♠ Simple regret minimization

♠ Pure exploration/best arm identification (BAI)

Fixed-confidence setting

Fixed-budget setting

Target of the agent

♠ **Cumulative regret minimization**

Maximize the cumulative reward, i.e.,minimize the regret (the gap between the maximum cumulative reward and the reward obtained by the agent) (Agrawal and Goyal, 2012; Russo and Van Roy, 2014; Lai, 1987).

♠ **Simple regret minimization**

Maximize the mean reward of the chosen arm by the end of a fixed time horizon T (Carpentier and Valko, 2015).

♠ **Pure exploration/best arm identification (BAI)**

Fixed-confidence setting

Fixed-budget setting

Target of the agent

♠ Cumulative regret minimization

Maximize the cumulative reward, i.e.,minimize the regret (the gap between the maximum cumulative reward and the reward obtained by the agent) (Agrawal and Goyal, 2012; Russo and Van Roy, 2014; Lai, 1987).

♠ Simple regret minimization

Maximize the mean reward of the chosen arm by the end of a fixed time horizon T (Carpentier and Valko, 2015).

♠ Pure exploration/best arm identification (BAI)

Fixed-confidence setting Given a risk parameter δ, the agent aims to identify the best arm with probability $1-\delta$ in minimal time steps (Jamieson and Nowak, 2014; Kalyanakrishnan et al., 2012).

Fixed-budget setting

Target of the agent

♠ Cumulative regret minimization

Maximize the cumulative reward, i.e.,minimize the regret (the gap between the maximum cumulative reward and the reward obtained by the agent) (Agrawal and Goyal, 2012; Russo and Van Roy, 2014; Lai, 1987).

♠ Simple regret minimization

Maximize the mean reward of the chosen arm by the end of a fixed time horizon T (Carpentier and Valko, 2015).

♠ Pure exploration/best arm identification (BAI)

Fixed-confidence setting Given a risk parameter δ, the agent aims to identify the best arm with probability $1-\delta$ in minimal time steps (Jamieson and Nowak, 2014; Kalyanakrishnan et al., 2012).

Fixed-budget setting Given a budget constraint T, the agent aims to maximize the confidence of the chosen arm by the end of a fixed time horizon T (Auer et al., 2002a; Audibert and Bubeck, 2010; Carpentier and Locatelli, 2016).

Outline

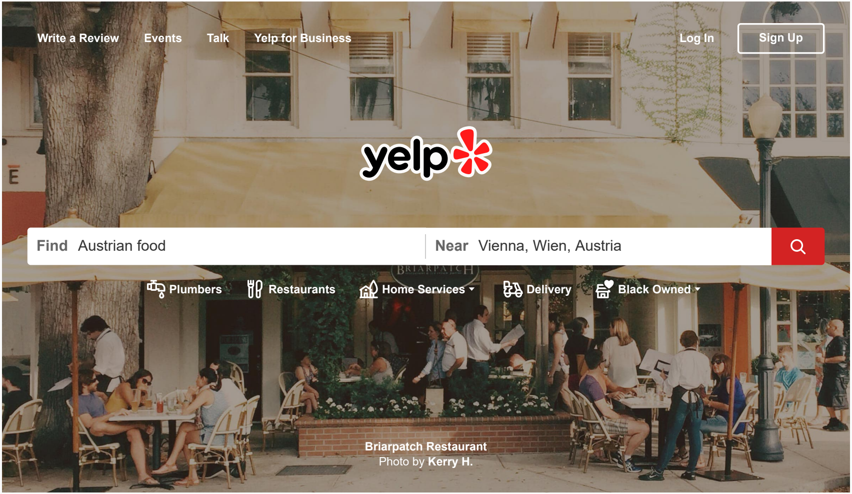
1. What is multi-armed bandits (MAB)?
   - Example — Cascading bandits
2. Explore state-of-the-art findings of pure exploration
3. Summary and discussions

♠ Online recommender system

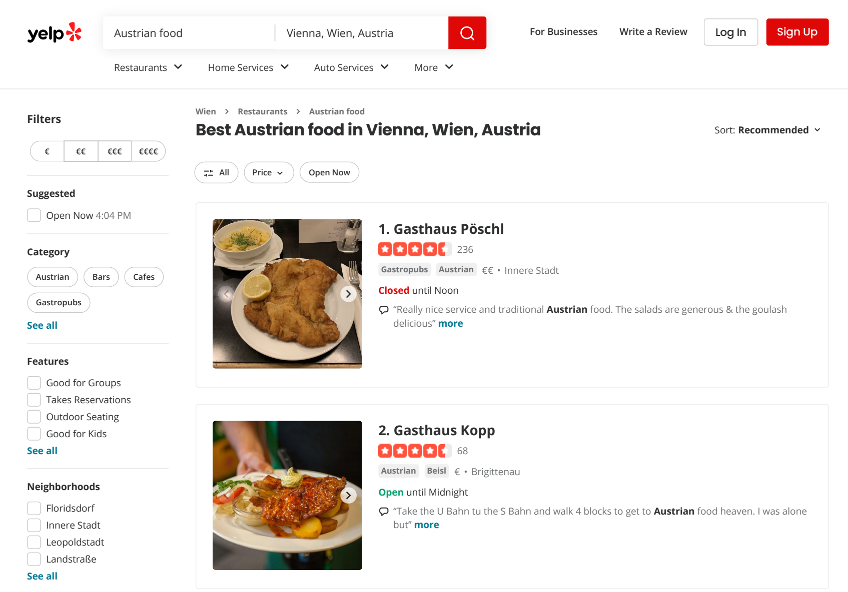
- seek to select a small list of items to the user over time.

♠ Online recommender system

- seek to select a small list of items to the user over time.



Example — Cascading bandits (Kveton et al., 2015a)

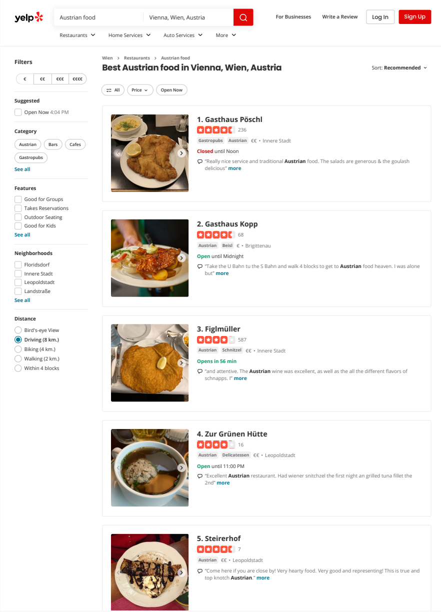


NUS National University of Singapore

♠ Online recommender system

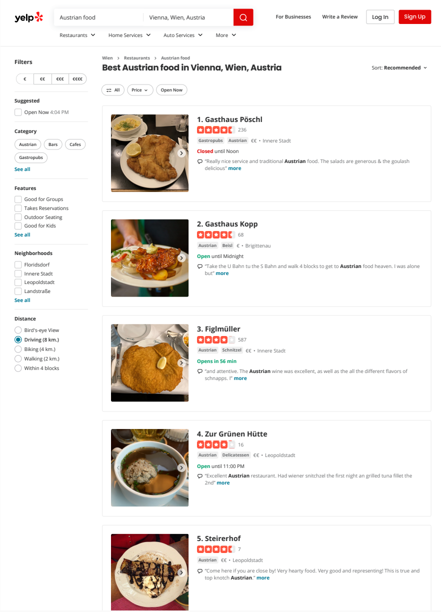
- Seek to select a small list of items to the user over time.
- How to maximize the 'reward' over several rounds of recommendation?

  — Regret Minimization (RM)



♠ **Online recommender system**

- Seek to select a small list of items to the user over time.
- How to maximize the 'reward' over several rounds of recommendation?
  — Regret Minimization (RM)
- How to select an attractive list of items after several rounds of recommendation?
  — Pure Exploration/
  Best Arm Identification (BAI)



Ground set

A finite set of all available arms $[L] := \{1, \ldots, L\}$.

Click probability/weight of item $i \in [L]$

Arm i attracts the user with probability $w(i) \in [0, 1]$.

Example — Cascading bandits (Kveton et al., 2015a)

Ground set

A finite set of all available arms $[L] := \{1, \ldots, L\}$.

Click probability/weight of item $i \in [L]$

Arm i attracts the user with probability $w(i) \in [0, 1]$.

- Standard setting: $\boldsymbol{w} := \{w(i)\}_{i=1}^{L}$ are not known.
- Linear generalization: $w(i) = x(i)^\top \beta$
  Feature vector $x(i)$ is known for each arm i, latent vector $\beta \in \mathbb{R}^d$ is not known.

Example — Cascading bandits (Kveton et al., 2015a)

Ground set

A finite set of all available arms $[L] := \{1, \ldots, L\}$.

Click probability/weight of item $i \in [L]$

Arm i attracts the user with probability $w(i) \in [0, 1]$.

- Standard setting: $\boldsymbol{w} := \{w(i)\}_{i=1}^{L}$ are not known.
- Linear generalization: $w(i) = x(i)^\top \beta$
  Feature vector $x(i)$ is known for each arm i, latent vector $\beta \in \mathbb{R}^d$ is not known.

Whether arm i is clicked at time t

This is revealed by a random variable $W_t(i) \sim \mathrm{Bern}\,(w(i))$.

- $W_t(i) = 1$ iff the user observes and clicks on i at time t.
- $W_t(i) = 0$ iff the user observes but does not click on i at time t.

Example — Cascading bandits (Kveton et al., 2015a)

Ground set

A finite set of all available arms $[L] := \{1, \ldots, L\}$.

Click probability/weight of item $i \in [L]$

Arm i attracts the user with probability $w(i) \in [0, 1]$.

- Standard setting: $\boldsymbol{w} := \{w(i)\}_{i=1}^{L}$ are not known.
- Linear generalization: $w(i) = x(i)^\top \beta$
  Feature vector $x(i)$ is known for each arm i, latent vector $\beta \in \mathbb{R}^d$ is not known.

Whether arm i is clicked at time t

This is revealed by a random variable $W_t(i) \sim \mathrm{Bern}\,(w(i))$.

- $W_t(i) = 1$ iff the user observes and clicks on i at time t.
- $W_t(i) = 0$ iff the user observes but does not click on i at time t.
- ♦ $W_t(i)$'s are only observed for some arms.

Example — Cascading bandits (Kveton et al., 2015a)

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

Example — Cascading bandits (Kveton et al., 2015a)

$$L = 9$$

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{\text{all } K\text{-permutations of } [L]\}$;

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation 

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation

$$K = 5$$

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

Recommendation

Attractiveness
$W_t(i)$

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation  

Attractiveness ×

$W_t(i)$ 0

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation			
Attractiveness	×	×	
$W_t(i)$	0	0	

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation

Attractiveness	×	×	×	
$W_t(i)$	0	0	0	

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

Recommendation

Attractiveness	$\times$	$\times$	$\times$	$\surd$
$W_t(i)$	0	0	0	1

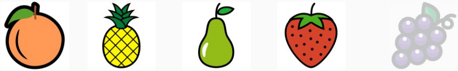
For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation



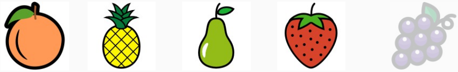
Attractiveness	$\times$	$\times$	$\times$	$\surd$	?
$W_t(i)$	0	0	0	1	?

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

Recommendation



Attractiveness	×	×	×	√	?
$W_t(i)$	0	0	0	1	?

For each time step $t = 1, 2, \ldots$

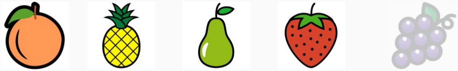
1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;

2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

♠ **Combinatorial** bandits ♡ **Partial** feedback

Example — Cascading bandits (Kveton et al., 2015a)

Recommendation



Attractiveness	$\times$	$\times$	$\times$	$\surd$	$?$
$W_t(i)$	0	0	0	1	$?$

For each time step $t = 1, 2, \ldots$

1. The agent selects a list of K arms $S_t := (i_1^t, \ldots, i_K^t) \in [L]^{(K)}$ to the user, where $[L]^{(K)} = \{$all K-permutations of $[L]\}$;
2. The user examines the arms from i_1^t to i_K^t:
   - If she is attracted by an item, clicks on it;
   - If not, she skips to the next item and checks if it is attractive;
   - Process stops when she clicks on one item or when she comes to the end of the list.

♠ **Combinatorial** bandits ♡ **Partial** feedback ♣ **Standard** setting & **Linear** generalization

Outline

1. What is multi-armed bandits (MAB)?
2. **Explore state-of-the-art findings of pure exploration**
3. Summary and discussions

Pure exploration/BAI settings

Fixed-confidence setting

- Given a risk parameter δ, the agent aims to identify the best arm with probability $1-\delta$ in minimal time steps.
  (Jamieson and Nowak, 2014; Kalyanakrishnan et al., 2012)

Fixed-budget setting

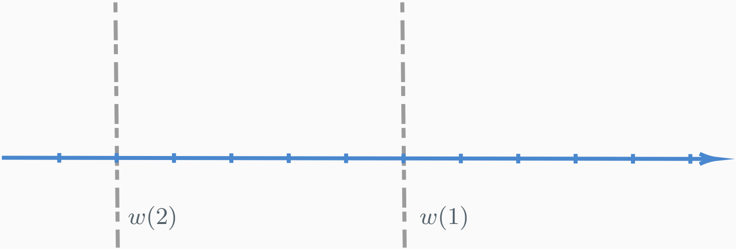
- Given a budget constraint T, the agent aims to maximize the confidence of the chosen arm by the end of a fixed time horizon T.
  (Auer et al., 2002a; Audibert and Bubeck, 2010; Carpentier and Locatelli, 2016)

Pure exploration in stochastic bandits

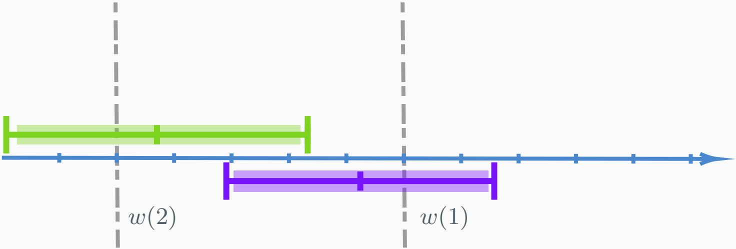
- Ground set $\mathcal{S} = [L]$ consists of L available arms.
- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.

Pure exploration in stochastic bandits

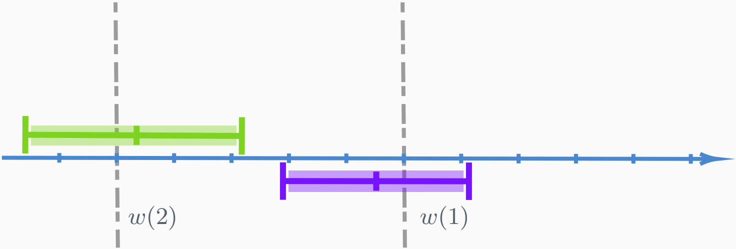
- Ground set $\mathcal{S} = [L]$ consists of L available arms.
- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.



- Ground set $\mathcal{S} = [L]$ consists of L available arms.
- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.



- Ground set $\mathcal{S} = [L]$ consists of L available arms.
- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma^2(i)$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.



- Optimal arm

$$1 = i^* = \arg\max_{i \in [L]} w(i)$$

  - Without loss of generality, assume

$$w(1) > w(2) \geq w(3) \geq \ldots \geq w(L).$$

- Optimal arm

$$1 = i^* = \arg\max_{i \in [L]} w(i)$$

  - Without loss of generality, assume

$$w(1) > w(2) \geq w(3) \geq \ldots \geq w(L).$$

- Gaps to optimality

$$\Delta_i = w(1) - w(i) \quad \forall\, i \neq 1, \quad \Delta_1 = \Delta_2.$$

Pure exploration in stochastic bandits

- Optimal arm

$$1 = i^* = \arg\max_{i \in [L]} w(i)$$

  - Without loss of generality, assume

$$w(1) > w(2) \geq w(3) \geq \ldots \geq w(L).$$

- Gaps to optimality

$$\Delta_i = w(1) - w(i) \quad \forall\, i \neq 1, \quad \Delta_1 = \Delta_2.$$

- Hardness parameters

$$H_1 = \sum_{i=1}^{L} \frac{1}{\Delta_i^2}, \quad H_2 = \max_{i \in [L]} \frac{i}{\Delta_i^2}.$$

Theorem 2.1 (Standard multiplicative variant of the Chernoff-Hoeffding bound; Dubhashi and Panconesi (2009), Theorem 1.1)

Suppose that $X_1, \ldots, X_T$ *are independent* $[0,1]$*-valued random variables, and let* $X = \sum_{t=1}^T X_t$*. Then for any* $\varepsilon \in (0,1)$*,*

$$\Pr(X - \mathbb{E}[X] \geq \varepsilon \mathbb{E}[X]) \leq \exp\left(-\frac{\varepsilon^2}{3}\mathbb{E}X\right),$$

$$\Pr(X - \mathbb{E}[X] \leq -\varepsilon \mathbb{E}[X]) \leq \exp\left(-\frac{\varepsilon^2}{3}\mathbb{E}X\right).$$

BAI algorithm

A deterministic and non-anticipatory online algorithm consists in a triple $\pi := ((\pi_t)_t, \mathcal{T}^\pi, \phi^\pi)$

- sampling rule $(\pi_t)_t$: which arm S_t^π to pull at time step t
- stopping rule $\mathcal{T}^\pi$: when to stop
- recommendation rule ϕ^π: which arm $\hat{S}^\pi$ to choose eventually

BAI algorithm

A deterministic and non-anticipatory online algorithm consists in a triple $\pi := ((\pi_t)_t, \mathcal{T}^\pi, \phi^\pi)$

- sampling rule $(\pi_t)_t$: which arm S_t^π to pull at time step t
  S_t^π is $\mathcal{F}_{t-1}$-measurable, observation history $\mathcal{F}_t := \sigma(S_1^\pi, \boldsymbol{O}_1^\pi, \ldots, S_t^\pi, \boldsymbol{O}_t^\pi)$;
- stopping rule $\mathcal{T}^\pi$: when to stop
- recommendation rule ϕ^π: which arm $\hat{S}^\pi$ to choose eventually

A deterministic and non-anticipatory online algorithm consists in a triple $\pi := ((\pi_t)_t, \mathcal{T}^\pi, \phi^\pi)$

- sampling rule $(\pi_t)_t$: which arm S_t^π to pull at time step t
  S_t^π is $\mathcal{F}_{t-1}$-measurable, observation history $\mathcal{F}_t := \sigma(S_1^\pi, \boldsymbol{O}_1^\pi, \ldots, S_t^\pi, \boldsymbol{O}_t^\pi)$;
- stopping rule $\mathcal{T}^\pi$: when to stop
  stopping time $\mathcal{T}^\pi$ with respect to $(\mathcal{F}_t)_{t\in\mathbb{N}}$ satisfying $\mathbb{P}(\mathcal{T}^\pi < \infty) = 1$;
- recommendation rule ϕ^π: which arm $\hat{S}^\pi$ to choose eventually

BAI algorithm

A deterministic and non-anticipatory online algorithm consists in a triple $\pi := ((\pi_t)_t, \mathcal{T}^\pi, \phi^\pi)$

- sampling rule $(\pi_t)_t$: which arm S_t^π to pull at time step t
  S_t^π is $\mathcal{F}_{t-1}$-measurable, observation history $\mathcal{F}_t := \sigma(S_1^\pi, O_1^\pi, \ldots, S_t^\pi, O_t^\pi)$;
- stopping rule $\mathcal{T}^\pi$: when to stop
  stopping time $\mathcal{T}^\pi$ with respect to $(\mathcal{F}_t)_{t\in\mathbb{N}}$ satisfying $\mathbb{P}(\mathcal{T}^\pi < \infty) = 1$;
- recommendation rule ϕ^π: which arm $\hat{S}^\pi$ to choose eventually
  $\mathcal{F}_{\mathcal{T}^\pi}$-measurable.

BAI algorithm

A deterministic and non-anticipatory online algorithm consists in a triple $\pi := ((\pi_t)_t, \mathcal{T}^\pi, \phi^\pi)$

- sampling rule $(\pi_t)_t$: which arm S_t^π to pull at time step t
  S_t^π is $\mathcal{F}_{t-1}$-measurable, observation history $\mathcal{F}_t := \sigma(S_1^\pi, \boldsymbol{O}_1^\pi, \ldots, S_t^\pi, \boldsymbol{O}_t^\pi)$;
- stopping rule $\mathcal{T}^\pi$: when to stop
  stopping time $\mathcal{T}^\pi$ with respect to $(\mathcal{F}_t)_{t \in \mathbb{N}}$ satisfying $\mathbb{P}(\mathcal{T}^\pi < \infty) = 1$;
- recommendation rule ϕ^π: which arm $\hat{S}^\pi$ to choose eventually
  $\mathcal{F}_{\mathcal{T}^\pi}$-measurable.

$\mathcal{T}^\pi$

- Fixed-confidence setting: Time complexity of π (to minimize).
- Fixed-budget setting: $\mathcal{T}^\pi = \mathcal{T}$ (fixed).

BAI: fixed-confidence

- δ-PAC algorithm: find the optimal arm with probability at least $1 - \delta$

BAI: fixed-confidence

- δ-PAC algorithm: find the optimal arm with probability at least $1 - \delta$

Theoretical study

▲ Propose a δ-PAC algorithm and upper bound its time complexity

▼ Derive a lower bound on the time complexity of any δ-PAC algorithm

- Evaluate theoretical findings with experiments

BAI: fixed-confidence

- δ-PAC algorithm: find the optimal arm with probability at least $1 - \delta$

Theoretical study

▲ Propose a δ-PAC algorithm and upper bound its time complexity
▼ Derive a lower bound on the time complexity of any δ-PAC algorithm
- Evaluate theoretical findings with experiments

Simple pure exploration in stochastic bandits

- to identify the best arm with the largest mean:

$$i^* = \arg\max_{i \in [L]} w(i)$$

BAI: fixed-confidence

- δ-PAC algorithm: find the optimal arm with probability at least $1-\delta$

Theoretical study

▲ Propose a δ-PAC algorithm and upper bound its time complexity
▼ Derive a lower bound on the time complexity of any δ-PAC algorithm
- Evaluate theoretical findings with experiments

Simple pure exploration in stochastic bandits

- to identify the best arm with the largest mean:

$$i^* = \arg\max_{i \in [L]} w(i)$$

♠ Successive elimination

SUCCESSIVE ELIMINATION, MEDIAN ELIMINATION (Even-Dar et al., 2002)

BAI: fixed-confidence

- δ-PAC algorithm: find the optimal arm with probability at least $1 - \delta$

Theoretical study

▲ Propose a δ-PAC algorithm and upper bound its time complexity
▼ Derive a lower bound on the time complexity of any δ-PAC algorithm
- Evaluate theoretical findings with experiments

Simple pure exploration in stochastic bandits

- to identify the best arm with the largest mean:

$$i^* = \arg\max_{i \in [L]} w(i)$$

♠ Successive elimination

SUCCESSIVE ELIMINATION, MEDIAN ELIMINATION (Even-Dar et al., 2002)

♠ Track optimal allocation

TRACK & STOP (Garivier and Kaufmann, 2016)

Algorithm 1: Successive Elimination(δ) (Even-Dar et al., 2002)

Algorithm 1: SUCCESSIVE ELIMINATION(δ) (Even-Dar et al., 2002)

Input: Set $t = 1$ and survival set $S = [L]$.
Let $\hat{w}_i^t$ be the average reward of arm i by time t.
Set $\hat{w}_i^1 = 0$ for all arm $i \in [L]$.

Algorithm 1: Successive Elimination(δ) (Even-Dar et al., 2002)

1: Input: Set $t = 1$ and survival set $S = [L]$.
2: Let $\hat{w}_i^t$ be the average reward of arm i by time t.
3: Set $\hat{w}_i^1 = 0$ for all arm $i \in [L]$.
4: Sample each arm $i \in S$ once and update $\hat{w}_i^t$ (average reward of arm i).
5: Let $\hat{w}_{\max}^t = \max_{i \in [L]} \hat{w}_i^t$ and confidence radius $\alpha_t = \sqrt{\frac{\log(cLt^2/\delta)}{t}}$.
6: For each arm $i \in S$ such that $\hat{w}_{\max}^t - \hat{w}_i^t \geq 2\alpha_t$, set $S = S \setminus \{i\}$.

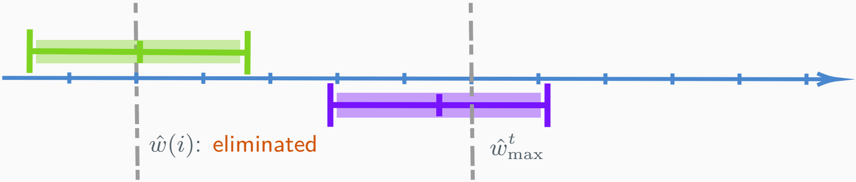
Algorithm 1: SUCCESSIVE ELIMINATION(δ) (Even-Dar et al., 2002)

1: Input: Set $t = 1$ and survival set $S = [L]$.
2: Let $\hat{w}_i^t$ be the average reward of arm i by time t.
3: Set $\hat{w}_i^1 = 0$ for all arm $i \in [L]$.
4: Sample each arm $i \in S$ once and update $\hat{w}_i^t$ (average reward of arm i).
5: Let $\hat{w}_{\max}^t = \max_{i \in [L]} \hat{w}_i^t$ and confidence radius $\alpha_t = \sqrt{\frac{\log(cLt^2/\delta)}{t}}$.
6: For each arm $i \in S$ such that $\hat{w}_{\max}^t - \hat{w}_i^t \geq 2\alpha_t$, set $S = S \setminus \{i\}$.
7: $t = t + 1$.
8: If $|S| > 1$, the Go to Step 4, Else output S.

Successive Elimination(δ) (Even-Dar et al., 2002)

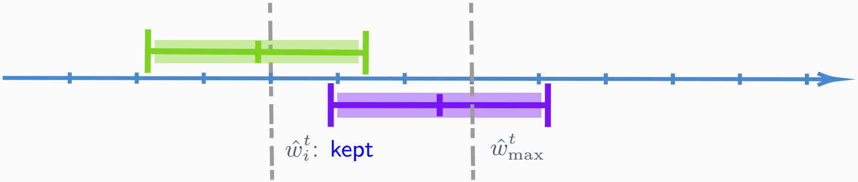
Algorithm 1: Successive Elimination(δ) (Even-Dar et al., 2002)

1: Input: Set $t = 1$ and survival set $S = [L]$.
2: Let $\hat{w}_i^t$ be the average reward of arm i by time t.
3: Set $\hat{w}_i^1 = 0$ for all arm $i \in [L]$.
4: Sample each arm $i \in S$ once and update $\hat{w}_i^t$ (average reward of arm i).
5: Let $\hat{w}_{\max}^t = \max_{i \in [L]} \hat{w}_i^t$ and confidence radius $\alpha_t = \sqrt{\frac{\log(cLt^2/\delta)}{t}}$.
6: For each arm $i \in S$ such that $\hat{w}_{\max}^t - \hat{w}_i^t \geq 2\alpha_t$, set $S = S \setminus \{i\}$.
7: $t = t + 1$.
8: If $|S| > 1$, the Go to Step 4, Else output S.



Algorithm 1: SUCCESSIVE ELIMINATION(δ) (Even-Dar et al., 2002)

1: Input: Set $t = 1$ and survival set $S = [L]$.
2: Let $\hat{w}_i^t$ be the average reward of arm i by time t.
3: Set $\hat{w}_i^1 = 0$ for all arm $i \in [L]$.
4: Sample each arm $i \in S$ once and update $\hat{w}_i^t$ (average reward of arm i).
5: Let $\hat{w}_{\max}^t = \max_{i \in [L]} \hat{w}_i^t$ and confidence radius $\alpha_t = \sqrt{\frac{\log(cLt^2/\delta)}{t}}$.
6: For each arm $i \in S$ such that $\hat{w}_{\max}^t - \hat{w}_i^t \geq 2\alpha_t$, set $S = S \setminus \{i\}$.
7: $t = t + 1$.
8: If $|S| > 1$, the Go to Step 4, Else output S.



Step 1. Concentration inequality:

$$\Pr\left(\cup_{i\in[L]} \cup_{t\in\mathbb{N}} \left\{ |\hat{w}_i^t - w(i)| > \alpha_t \right\} \right) \le \delta.$$

Step 1. Concentration inequality:

$$\Pr\left(\cup_{i\in[L]} \cup_{t\in\mathbb{N}} \left\{ |\hat{w}_i^t - w(i)| > \alpha_t \right\} \right) \le \delta.$$

Step 2. Assume $\cap_{i\in[L]} \cap_{t\in\mathbb{N}} \left\{ |\hat{w}_i^t - w(i)| \le \alpha_t \right\}$ holds. Recall that we eliminate arm i when $\hat{w}_{\max}^t - \hat{w}_i^t \ge 2\alpha_t$. Since

$$\hat{w}_{\max}^t - \hat{w}_i^t \ge \hat{w}_1^t - \hat{w}_i^t \ge w(1) - \alpha_t - (w(i) + \alpha_t) = w(1) - w(i) - 2\alpha_2,$$

we eliminate a suboptimal arm $i \ne 1$ when

$$w(1) - w(i) - 2\alpha_t = \Delta_i - 2\alpha_t \ge 2\alpha_t.$$

Step 1. Concentration inequality:

$$\Pr\left(\cup_{i\in[L]}\cup_{t\in\mathbb{N}}\left\{|\hat{w}_i^t - w(i)| > \alpha_t\right\}\right) \le \delta.$$

Step 2. Assume $\cap_{i\in[L]}\cap_{t\in\mathbb{N}}\left\{|\hat{w}_i^t - w(i)| \le \alpha_t\right\}$ holds. Recall that we eliminate arm i when $\hat{w}_{\max}^t - \hat{w}_i^t \ge 2\alpha_t$. Since

$$\hat{w}_{\max}^t - \hat{w}_i^t \ge \hat{w}_1^t - \hat{w}_i^t \ge w(1) - \alpha_t - (w(i) + \alpha_t) = w(1) - w(i) - 2\alpha_2,$$

we eliminate a suboptimal arm $i \neq 1$ when

$$w(1) - w(i) - 2\alpha_t = \Delta_i - 2\alpha_t \ge 2\alpha_t.$$

Step 3. When each arm has been sampled for

$$t_i = O\left(\frac{\log(L/(\delta\Delta_i))}{\Delta_i^2}\right)$$

times, we have $\alpha_t \le \Delta_i/4$ and arm i will be eliminated.

Hence, the time complexity would be

$$t_2 + \sum_{i=2}^{L} t_i = O\left(\sum_{t=1}^{L}\frac{\log(L/(\delta\Delta_i))}{\Delta_i^2}\right) = \tilde{O}(H_1), \quad H_1 = \sum_{i=1}^{L}\frac{1}{\Delta_i^2} \text{ (hardness)}.$$

♠ With probability $1 - \delta$, identify an ϵ-optimal arm i: $w(i) \geq \max_{j \in [L]} w(j) - \epsilon$.

Algorithm 2: MEDIAN ELIMINATION(ϵ, δ) (Even-Dar et al., 2002)

MEDIAN ELIMINATION(ϵ, δ) **(Even-Dar et al., 2002)**

♠ With probability $1 - \delta$, identify an ϵ-optimal arm i: $w(i) \geq \max_{j \in [L]} w(j) - \epsilon$.

Algorithm 2: MEDIAN ELIMINATION(ϵ, δ) (Even-Dar et al., 2002)

1: Input: Survival set $S = [L]$. Set $\epsilon_1 = \epsilon/4$, $\delta_1 = \delta/2$, $\ell = 1$.

2: Sample each arm $i \in S$ for $\frac{1}{(\epsilon_\ell/2)^2} \log(3/\delta_\ell)$ times, and let $\hat{w}_i^t$ denote its average reward.

♠ With probability $1 - \delta$, identify an ϵ-optimal arm i: $w(i) \geq \max_{j \in [L]} w(j) - \epsilon$.

Algorithm 2: MEDIAN ELIMINATION(ϵ, δ) (Even-Dar et al., 2002)

1: Input: Survival set $S = [L]$. Set $\epsilon_1 = \epsilon/4$, $\delta_1 = \delta/2$, $\ell = 1$.

2: Sample each arm $i \in S$ for $\dfrac{1}{(\epsilon_\ell/2)^2} \log(3/\delta_\ell)$ times, and let $\hat{w}_i^t$ denote its average reward.

3: Find the median of $\hat{w}_i^\ell$, denoted by $m_\ell :=$ median$\big(\{\hat{w}_i^\ell : i \in S_\ell\}\big)$.

4: Let $S_{\ell+1} = S_\ell \setminus \{i : \hat{w}_i^\ell < m_\ell\}$.

MEDIAN ELIMINATION(ϵ, δ) **(Even-Dar et al., 2002)**

♠ With probability $1 - \delta$, identify an ϵ-optimal arm i: $w(i) \geq \max_{j \in [L]} w(j) - \epsilon$.

Algorithm 2: MEDIAN ELIMINATION(ϵ, δ) (Even-Dar et al., 2002)

1: Input: Survival set $S = [L]$. Set $\epsilon_1 = \epsilon/4$, $\delta_1 = \delta/2$, $\ell = 1$.
2: Sample each arm $i \in S$ for $\dfrac{1}{(\epsilon_\ell/2)^2} \log(3/\delta_\ell)$ times, and let $\hat{w}_i^t$ denote its average reward.
3: Find the median of $\hat{w}_i^\ell$, denoted by $m_\ell :=$ median$\big(\{\hat{w}_i^\ell : i \in S_\ell\}\big)$.
4: Let $S_{\ell+1} = S_\ell \setminus \{i : \hat{w}_i^\ell < m_\ell\}$.
5: $t = t + 1$.
6: If $|S| = 1$, Then output S.
Else $\epsilon_{\ell+1} = \frac{3}{4}\epsilon_\ell$, $\delta_{\ell+1} = \delta_\ell/2$, $\ell = \ell + 1$; Go to Step 2.

♠ With probability $1 - \delta$, identify an ϵ-optimal arm i: $w(i) \geq \max_{j \in [L]} w(j) - \epsilon$.

Algorithm 2: MEDIAN ELIMINATION(ϵ, δ) (Even-Dar et al., 2002)

1: Input: Survival set $S = [L]$. Set $\epsilon_1 = \epsilon/4$, $\delta_1 = \delta/2$, $\ell = 1$.

2: Sample each arm $i \in S$ for $\frac{1}{(\epsilon_\ell/2)^2} \log(3/\delta_\ell)$ times, and let $\hat{w}_i^t$ denote its average reward.

3: Find the median of $\hat{w}_i^\ell$, denoted by $m_\ell :=$ median$\big(\{\hat{w}_i^\ell : i \in S_\ell\}\big)$.

4: Let $S_{\ell+1} = S_\ell \setminus \{i : \hat{w}_i^\ell < m_\ell\}$.

5: $t = t + 1$.

6: If $|S| = 1$, Then output S.
Else $\epsilon_{\ell+1} = \frac{3}{4}\epsilon_\ell$, $\delta_{\ell+1} = \delta_\ell/2$, $\ell = \ell + 1$; Go to Step 2.

Applying the same concentration inequality, we can show the time complexity of MEDIAN ELIMINATION(ϵ, δ) is

$$O\left(\frac{L \log(1/\delta)}{\epsilon^2}\right).$$

Lower bound (Garivier and Kaufmann, 2016)

For any δ-PAC algorithm and any bandit instance μ,

$$\mathbb{E}_\mu[\tau_\delta] \geq T^*(\mu) \log\left(\frac{4}{\delta}\right)$$

where

$$T^*(\mu)^{-1} := \sup_{w \in \Sigma_L} \inf_{\lambda \in \mathrm{Alt}(\mu)} \left(\sum_{i=1}^{L} w_i d(\mu_i, \lambda_i) \right).$$

Lower bound (Garivier and Kaufmann, 2016)

For any δ-PAC algorithm and any bandit instance μ,

$$\mathbb{E}_\mu[\tau_\delta] \geq T^*(\mu) \log\left(\frac{4}{\delta}\right)$$

where

$$T^*(\mu)^{-1} := \sup_{w \in \Sigma_L} \inf_{\lambda \in \mathrm{Alt}(\mu)} \left(\sum_{i=1}^{L} w_i d(\mu_i, \lambda_i) \right).$$

- For any instance $\mu = (\mu_1, \ldots, \mu_L) \in \mathcal{S}$
  - $\mathcal{S} = \{(\mu_1, \ldots, \mu_L) : \exists i^*(\mu) \in [L] \ s.t. \ \mu_{i^*(\mu)} > \mu_i \quad \forall i \neq i^*(\mu)\}$
  - Unique optimal arm: $i^*(\mu) = \arg\max_{i \in [L]} \mu_i$
  - "Alternative set": $\mathrm{Alt}(\mu) := \{\lambda \in \mathcal{S} : i^*(\lambda) \neq i^*(\mu)\}$
- Set of probability distributions on $[L]$

$$\Sigma_L = \left\{ (w_1, \ldots, w_L) \in (0,1]^L : \sum_{i=1}^{L} w_i = 1 \right\}$$

Proof strategy of lower bound

- Let $\lambda \in \mathrm{Alt}(\mu)$ and define event $E = \{\tau_\delta < \infty, i_{\mathrm{out}}(\mu) \neq i^*(\lambda)\} \in \mathcal{F}_{\tau_\delta}$. Then

$$2\delta \geq \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\mu)) + \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\lambda))$$
$$\geq \mathbb{P}_\mu(E^c) + \mathbb{P}_\lambda(E)$$
$$\geq \frac{1}{2} \exp\left(- \sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \right). \qquad \text{Bretagnolle–Huber inequality}$$

Proof strategy of lower bound

- Let $\lambda \in \mathrm{Alt}(\mu)$ and define event $E = \{\tau_\delta < \infty, i_{\mathrm{out}}(\mu) \neq i^*(\lambda)\} \in \mathcal{F}_{\tau_\delta}$. Then

$$
\begin{aligned}
2\delta &\geq \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\mu)) + \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\lambda)) \\
&\geq \mathbb{P}_\mu(E^c) + \mathbb{P}_\lambda(E) \\
&\geq \frac{1}{2} \exp\left(- \sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \right). \qquad \text{Bretagnolle–Huber inequality}
\end{aligned}
$$

- Rearranging,

$$
\sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \geq \log \frac{4}{\delta}
$$

Proof strategy of lower bound

- Let $\lambda \in \mathrm{Alt}(\mu)$ and define event $E = \{\tau_\delta < \infty, i_{\mathrm{out}}(\mu) \neq i^*(\lambda)\} \in \mathcal{F}_{\tau_\delta}$. Then

$$
\begin{aligned}
2\delta &\geq \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\mu)) + \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\lambda)) \\
&\geq \mathbb{P}_\mu(E^c) + \mathbb{P}_\lambda(E) \\
&\geq \frac{1}{2} \exp\left(- \sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \right). \qquad \text{Bretagnolle–Huber inequality}
\end{aligned}
$$

- Rearranging,

$$
\sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \geq \log \frac{4}{\delta}
$$

- Using this and the definition of $T^*(\mu)$,

$$
\begin{aligned}
\frac{\mathbb{E}_\mu[\tau_\delta]}{T^*(\mu)} &= \mathbb{E}_\mu[\tau_\delta] \sup_{w \in \Sigma_L} \inf_{\lambda \in \mathrm{Alt}(\mu)} \sum_{i=1}^{L} w_i D(\mu_i, \lambda_i) \\
&\geq \cancel{\mathbb{E}_\mu[\tau_\delta]} \inf_{\lambda \in \mathrm{Alt}(\mu)} \sum_{i=1}^{L} \frac{\mathbb{E}_\mu[T_i(\tau_\delta)]}{\cancel{\mathbb{E}_\mu[\tau_\delta]}} D(\mu_i, \lambda_i) \\
&= \inf_{\lambda \in \mathrm{Alt}(\mu)} \sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \geq \log \frac{4}{\delta}.
\end{aligned}
$$

Proof strategy of lower bound

- Let $\lambda \in \mathrm{Alt}(\mu)$ and define event $E = \{\tau_\delta < \infty, i_{\mathrm{out}}(\mu) \neq i^*(\lambda)\} \in \mathcal{F}_{\tau_\delta}$. Then

$$\begin{aligned} 2\delta &\ge \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\mu)) + \mathbb{P}_\mu(\tau_\delta < \infty \text{ and } i_{\mathrm{out}}(\mu) \neq i^*(\lambda)) \\ &\ge \mathbb{P}_\mu(E^c) + \mathbb{P}_\lambda(E) \\ &\ge \frac{1}{2} \exp\left(- \sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \right). \qquad \text{Bretagnolle–Huber inequality} \end{aligned}$$

- Rearranging,

$$\sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \ge \log \frac{4}{\delta}$$

- Using this and the definition of $T^*(\mu)$,

$$\begin{aligned} \frac{\mathbb{E}_\mu[\tau_\delta]}{T^*(\mu)} &= \mathbb{E}_\mu[\tau_\delta] \sup_{w \in \Sigma_L} \inf_{\lambda \in \mathrm{Alt}(\mu)} \sum_{i=1}^{L} w_i D(\mu_i, \lambda_i) \\ &\ge \cancel{\mathbb{E}_\mu[\tau_\delta]} \inf_{\lambda \in \mathrm{Alt}(\mu)} \sum_{i=1}^{L} \frac{\mathbb{E}_\mu[T_i(\tau_\delta)]}{\cancel{\mathbb{E}_\mu[\tau_\delta]}} D(\mu_i, \lambda_i) \\ &= \inf_{\lambda \in \mathrm{Alt}(\mu)} \sum_{i=1}^{L} \mathbb{E}_\mu[T_i(\tau_\delta)] D(\mu_i, \lambda_i) \ge \log \frac{4}{\delta}. \end{aligned}$$

We thus have the asymptotic lower bound on the time complexity:

$$\liminf_{\delta \to 0} \frac{\mathbb{E}_{\mu}[\tau_{\delta}]}{\log(1/\delta)} \geq T^{*}(\mu).$$

Track & Stop algorithm (Garivier and Kaufmann, 2016)

We thus have the asymptotic lower bound on the time complexity:

$$\liminf_{\delta\to 0} \frac{\mathbb{E}_\mu[\tau_\delta]}{\log(1/\delta)} \geq T^*(\mu).$$

A matching upper bound can be achieved by Track & Stop

$$\mathbb{P}_\mu\left(\limsup_{\delta\to 0} \frac{\tau_\delta}{\log(1/\delta)} \leq T^*(\mu)\right) = 1,$$

or

$$\limsup_{\delta\to 0} \frac{\mathbb{E}_\mu[\tau_\delta]}{\log(1/\delta)} \leq T^*(\mu).$$

Algorithm 3: Track & Stop (Garivier and Kaufmann, 2016)

1: Let $N_i(t) = \sum_{u=1}^{t} 1\{S_u = i\}$ be the number of pulls of arm i,

$$\hat{\mu}_i(t) = \frac{1}{N_i(t)} \sum_{u=1}^{t} W_t(i) 1\{S_u = i\}$$ be the empirical mean of arm i.

Set $\hat{\mu}(t) = (\hat{\mu}_1(t), \hat{\mu}_2(t), \ldots, \hat{\mu}_L(t))$.

2: Sample each arm once and update $t = L$, $N_i(L)$, $\hat{\mu}_i(L)$.

Algorithm 3: TRACK & STOP (Garivier and Kaufmann, 2016)

1: Let $N_i(t) = \sum_{u=1}^{t} 1\{S_u = i\}$ be the number of pulls of arm i,

$$\hat{\mu}_i(t) = \frac{1}{N_i(t)} \sum_{u=1}^{t} W_t(i) 1\{S_u = i\}$$ be the empirical mean of arm i.

Set $\hat{\mu}(t) = (\hat{\mu}_1(t), \hat{\mu}_2(t), \ldots, \hat{\mu}_L(t))$.

2: Sample each arm once and update $t = L$, $N_i(L)$, $\hat{\mu}_i(L)$.

3: **while** *Stopping condition (Generalized Likelihood Ratio statistic)* is not satisfied **do**

4: Sample arm S_{t+1} by C-Tracking/D-Tracking rule.

5: Let $t = t + 1$, and update $N_i(t)$, $\hat{\mu}_i(t)$.

6: **end while**

7: Output $\hat{i} = \arg\max_{i \in [L]} \hat{\mu}_i(t)$.

Sampling rule

C-Tracking: $S_{t+1} \in \arg\max_{i \in [L]} \sum_{\tau=0}^{t} w_i^{\epsilon_\tau}(\hat{\mu}(\tau)) - N_i(t)$

D-Tracking: $S_{t+1} \in \begin{cases} \arg\min_{i \in U_t} N_i(t) & \text{if } U_t \neq \emptyset \quad \text{(forced exploration)} \\ \arg\max_{i \in [L]} t w_i^{\epsilon_t}(\hat{\mu}(t)) - N_i(t) & \text{else} \quad \text{(directed tracking)} \end{cases}$

Sampling rule

C-Tracking: $S_{t+1} \in \arg\max_{i\in[L]} \sum_{\tau=0}^{t} w_i^{\epsilon_\tau}(\hat{\mu}(\tau)) - N_i(t)$

D-Tracking: $S_{t+1} \in \begin{cases} \arg\min_{i\in U_t} N_i(t) & \text{if } U_t \neq \emptyset \quad \text{(forced exploration)} \\ \arg\max_{i\in[L]} t w_i^{\epsilon_t}(\hat{\mu}(t)) - N_i(t) & \text{else} \quad \text{(directed tracking)} \end{cases}$

$$w^*(\mu) = \arg\max_{w\in\Sigma_L} \inf_{\lambda\in\mathrm{Alt}(\mu)} \left(\sum_{i=1}^{L} w_i d(w_i, \lambda_i)\right),$$

- Proportion of arm draws of any strategy matches the lower bound

$\epsilon_t = (L^2 + t)^{-1/2}/2$,

$w^\epsilon(\mu)$: L^∞ projection of $w^*(\mu)$ onto $\Sigma_L^{(\epsilon)} = \left\{(w_1, \ldots, w_L) \in [\epsilon, 1]^L : \sum_{i=1}^{L} = 1\right\}$

Sampling rule

C-Tracking: $S_{t+1} \in \arg\max_{i\in[L]} \sum_{\tau=0}^{t} w_i^{\epsilon_\tau}(\hat{\mu}(\tau)) - N_i(t)$

D-Tracking: $S_{t+1} \in \begin{cases} \arg\min_{i\in U_t} N_i(t) & \text{if } U_t \neq \emptyset \quad \text{(forced exploration)} \\ \arg\max_{i\in[L]} t w_i^{\epsilon_t}(\hat{\mu}(t)) - N_i(t) & \text{else} \quad \text{(directed tracking)} \end{cases}$

$$w^*(\mu) = \arg\max_{w\in\Sigma_L} \inf_{\lambda\in\mathrm{Alt}(\mu)} \left(\sum_{i=1}^{L} w_i d(w_i, \lambda_i) \right),$$

- Proportion of arm draws of any strategy matches the lower bound

$\epsilon_t = (L^2 + t)^{-1/2}/2$,

$w^\epsilon(\mu)$: L^∞ projection of $w^*(\mu)$ onto $\Sigma_L^{(\epsilon)} = \left\{ (w_1, \ldots, w_L) \in [\epsilon, 1]^L : \sum_{i=1}^{L} = 1 \right\}$

Outline

BAI: fixed-budget

Theoretical study

▲ Propose a BAI algorithm in a fixed time horizon and upper bound its failure probability

▼ Derive a lower bound on the failure probability of any algorithm

- Evaluate theoretical findings with experiments

BAI: fixed-budget

Theoretical study

▲ Propose a BAI algorithm in a fixed time horizon and upper bound its failure probability

▼ Derive a lower bound on the failure probability of any algorithm

- Evaluate theoretical findings with experiments

Simple pure exploration in stochastic bandits

- to identify the best arm with the largest mean: $i^* = \arg\max_{i \in [L]} w(i)$

BAI: fixed-budget

Theoretical study

▲ Propose a BAI algorithm in a fixed time horizon and upper bound its failure probability

▼ Derive a lower bound on the failure probability of any algorithm

- Evaluate theoretical findings with experiments

Simple pure exploration in stochastic bandits

- to identify the best arm with the largest mean: $i^* = \arg\max_{i \in [L]} w(i)$

♠ UCB-based

$\mathrm{UCB\text{-}E}(a)$ (Audibert and Bubeck, 2010)

♠ Successive elimination

SEQUENTIAL HALVING (Karnin et al., 2013)

UCB-E(a) (Audibert and Bubeck, 2010)

Algorithm 4: UCB-E(a) (Audibert and Bubeck, 2010)

Input: time budget T, size of ground set of items L, parameter a.

For all $i \in [L]$, compute $N_{i,0}$, $\hat{w}_{i,0}$, $C_{i,0}$, $U_{i,0}$:

$$N_{i,t} = \sum_{u=1}^{t} 1\{i_u = i\}, \;\; \hat{w}_{i,t} = \frac{1}{N_{i,t}} \sum_{u=1}^{t} W_{i,t} \cdot 1\{i_u = i\},$$

$$C_{i,t} = \sqrt{\frac{a}{t}} \text{ if } t \ge 1, \qquad C_{i,0} = +\infty, \qquad U_{i,t} = \hat{g}_{i,t} + C_{i,t}.$$

for $t = 1, \ldots, T$ **do**

  Pull item $i_t = \arg\max_{i \in [L]} U_{i,t-1}$.

  Update $N_{i_t,t}$, $\hat{w}_{i_t,t}$, $C_{i,t}$, and $U_{i,t}$ for all i.

end for

Output $i_{\text{out}} = \arg\max_{i \in [L]} \hat{w}_{i,T}$.

Step 1: Concentration. Let $\mathcal{E}_i := \{\forall t \geq L, |\hat{w}_{i,t} - w(i)| \leq C_{i,t}/5\}$ for all $i \in [L]$. We apply concentration inequality to show that

$$\Pr\left(\bigcap_{i=1}^{L} \mathcal{E}_i\right) \geq 1 - 2TL \exp\left(-\frac{2a}{25}\right).$$

In the following, we prove that conditioned on the event $\bigcap_{i=1}^{L} \mathcal{E}_i$, we have $i_{\text{out}} = 1$, which concludes the proof.

We assume $\bigcap_{i=1}^{L} \mathcal{E}_i$ holds from now on. Since i_{out} is the item with the largest empirical mean, for all $i \neq i_{\text{out}}$, we have

$$\hat{w}_{i_{\text{out}},T} \geq \hat{w}_{i,t}, \quad \hat{w}_{i_{\text{out}},T} \geq w(i_{\text{out}}) - C_{i_{\text{out}},T}/5, \quad w(i) + C_{i,T}/5 \geq \hat{w}_{i,t}.$$

Consequently, to show $i_{\text{out}} = 1$, it is sufficient to show that

$$\frac{C_{i,T}}{5} \leq \frac{\Delta_i}{2} \Leftrightarrow N_{it} \geq \frac{4}{25}\frac{a}{\Delta_i^2} \quad \forall i \in [L]. \tag{1}$$

Step 1: Concentration. Let $\mathcal{E}_i := \{\forall t \geq L, |\hat{w}_{i,t} - w(i)| \leq C_{i,t}/5\}$ for all $i \in [L]$. We apply concentration inequality to show that

$$\Pr\left(\bigcap_{i=1}^{L} \mathcal{E}_i\right) \geq 1 - 2TL\exp\left(-\frac{2a}{25}\right).$$

In the following, we prove that conditioned on the event $\bigcap_{i=1}^{L} \mathcal{E}_i$, we have $i_{\text{out}} = 1$, which concludes the proof.

We assume $\bigcap_{i=1}^{L} \mathcal{E}_i$ holds from now on. Since i_{out} is the item with the largest empirical mean, for all $i \neq i_{\text{out}}$, we have

$$\hat{w}_{i_{\text{out}},T} \geq \hat{w}_{i,t}, \quad \hat{w}_{i_{\text{out}},T} \geq w(i_{\text{out}}) - C_{i_{\text{out}},T}/5, \quad w(i) + C_{i,T}/5 \geq \hat{w}_{i,t}.$$

Consequently, to show $i_{\text{out}} = 1$, it is sufficient to show that

$$\frac{C_{i,T}}{5} \leq \frac{\Delta_i}{2} \;\Leftrightarrow\; N_{it} \geq \frac{4}{25}\frac{a}{\Delta_i^2} \quad \forall i \in [L]. \tag{1}$$

Step 2: Upper bound $N_{i,T}$ $(i \neq 1)$. To begin with, we prove by induction that

$$N_{i,t} \leq \frac{36}{25}\frac{a}{\Delta_i^2} \quad \forall i \neq 1. \tag{2}$$

Step 3: Lower bound $N_{i,T}$ $(i \neq 1)$. Next, we again prove by induction that

$$N_{i,t} \geq \frac{4}{25} \min\left\{ \frac{a}{\Delta_i^2}, \frac{25}{36}(N_{1,t} - 1) \right\} \quad \forall i \neq 1. \tag{3}$$

Step 3: Lower bound $N_{i,T}$ $(i \neq 1)$. Next, we again prove by induction that

$$N_{i,t} \geq \frac{4}{25} \min\left\{\frac{a}{\Delta_i^2}, \frac{25}{36}(N_{1,t}-1)\right\} \quad \forall i \neq 1. \qquad (3)$$

Step 4: Lower bound on $N_{1,T}$. Recall that we want to show (1). (i) To show (1) holds for all $i \neq 1$, (3) indicates that it is sufficient to show that

$$\frac{25}{36}(N_{1,t}-1) \geq \frac{a}{\Delta_i^2} \quad \forall i \neq 1.$$

(ii) In order to show (1) holds for all $i = 1$, we apply (2), $t = \sum_{i=1}^{L} N_{i,t}$ and

$$\frac{36}{25} H_1 a \leq T - L \ \Leftrightarrow a \leq \frac{25(T-L)}{36 H_1}, \quad H_1 = \sum_{i=1}^{L} \frac{1}{\Delta_i^2}.$$

Step 3: Lower bound $N_{i,T}$ $(i \neq 1)$. Next, we again prove by induction that

$$N_{i,t} \geq \frac{4}{25} \min\left\{\frac{a}{\Delta_i^2}, \frac{25}{36}(N_{1,t}-1)\right\} \quad \forall i \neq 1. \tag{3}$$

Step 4: Lower bound on $N_{1,T}$. Recall that we want to show (1). (i) To show (1) holds for all $i \neq 1$, (3) indicates that it is sufficient to show that

$$\frac{25}{36}(N_{1,t}-1) \geq \frac{a}{\Delta_i^2} \quad \forall i \neq 1.$$

(ii) In order to show (1) holds for all $i = 1$, we apply (2), $t = \sum_{i=1}^{L} N_{i,t}$ and

$$\frac{36}{25} H_1 a \leq T - L \;\Leftrightarrow a \leq \frac{25(T-L)}{36 H_1}, \quad H_1 = \sum_{i=1}^{L} \frac{1}{\Delta_i^2}.$$

Step 5: Conclusion. The failure probability is

$$2TL \exp\left(-\frac{2a}{25}\right) \quad \forall\, a \leq \frac{25(T-L)}{36H_1}$$

and achieves the minimum,

$$2TL \exp\left(-\frac{T-L}{18H_1}\right) \quad \text{when } a = \frac{25(T-L)}{36H_1}.$$

Step 3: Lower bound $N_{i,T}$ $(i \neq 1)$. Next, we again prove by induction that

$$N_{i,t} \geq \frac{4}{25} \min \left\{ \frac{a}{\Delta_i^2}, \frac{25}{36}(N_{1,t} - 1) \right\} \quad \forall i \neq 1. \tag{3}$$

Step 4: Lower bound on $N_{1,T}$. Recall that we want to show (1). (i) To show (1) holds for all $i \neq 1$, (3) indicates that it is sufficient to show that

$$\frac{25}{36}(N_{1,t} - 1) \geq \frac{a}{\Delta_i^2} \quad \forall i \neq 1.$$

(ii) In order to show (1) holds for all $i = 1$, we apply (2), $t = \sum_{i=1}^{L} N_{i,t}$ and

$$\frac{36}{25} H_1 a \leq T - L \ \Leftrightarrow a \leq \frac{25(T-L)}{36 H_1}, \quad H_1 = \sum_{i=1}^{L} \frac{1}{\Delta_i^2}.$$

Step 5: Conclusion. The failure probability is

$$2TL \exp\left(-\frac{2a}{25}\right) \quad \forall\, a \leq \frac{25(T-L)}{36 H_1}$$

and achieves the minimum, however, requiring prior knowledge: hardness H_1

$$2TL \exp\left(-\frac{T-L}{18 H_1}\right) \quad \text{when } a = \frac{25(T-L)}{36 H_1}.$$

Algorithm 5: Sequential Halving (SH) (Karnin et al., 2013)

1: Input: time budget T, size of ground set L.
2: Set $M = \lceil \log_2 L \rceil$, $N = \lfloor T/M \rfloor$, $T_0 = 0$, $A_0 = [L]$.

Algorithm 5: Sequential Halving (SH) (Karnin et al., 2013)

1: Input: time budget T, size of ground set L.
2: Set $M = \lceil \log_2 L \rceil$, $N = \lfloor T/M \rfloor$, $T_0 = 0$, $A_0 = [L]$.

M : number of phases

N : length of each phase

T_m : last time step of phase m

A_m : active set after phase m

Sequantial Halving (SH) **(Karnin et al., 2013)**

Algorithm 5: Sequantial Halving (SH) (Karnin et al., 2013)

Input: time budget T, size of ground set L.
Set $M = \lceil \log_2 L \rceil$, $N = \lfloor T/M \rfloor$, $T_0 = 0$, $A_0 = [L]$.
for phase $m = 1, 2, \ldots, M$ **do**
    Set $T_m = T_{m-1} + N, q_m = 1/|A_{m-1}|, n_m = \lfloor q_m N \rfloor$.
    for $t = T_{m-1} + 1, \ldots, T_m$ **do**
        Pull $i \in A_{m-1}$ with for n_m times in order and observe $W_t(i)$.
    end for

Sequantial Halving (SH) (Karnin et al., 2013)

Algorithm 5: Sequantial Halving (SH) (Karnin et al., 2013)

Input: time budget T, size of ground set L.
Set $M = \lceil \log_2 L \rceil$, $N = \lfloor T/M \rfloor$, $T_0 = 0$, $A_0 = [L]$.
for phase $m = 1, 2, \ldots, M$ **do**
    Set $T_m = T_{m-1} + N, q_m = 1/|A_{m-1}|, n_m = \lfloor q_m N \rfloor$.
    for $t = T_{m-1} + 1, \ldots, T_m$ **do**
        Pull $i \in A_{m-1}$ with for n_m times in order and observe $W_t(i)$.
    end for
    For all $i \in A_{m-1}$, set

$$S_m(i) = \sum_{t=T_{m-1}+1}^{T_m} W_t(i_t) \cdot \mathbb{I}\{i_t = i\}, \ \hat{w}_m(i) = \frac{S_m(i)}{n_m}.$$

    Let A_m contain the $\lceil L/2^m \rceil$ items with the highest $\hat{w}_m(i)$'s in A_{m-1}.

Sequantial Halving (SH) **(Karnin et al., 2013)**

Algorithm 5: Sequantial Halving (SH) (Karnin et al., 2013)

Input: time budget T, size of ground set L.
Set $M = \lceil \log_2 L \rceil$, $N = \lfloor T/M \rfloor$, $T_0 = 0$, $A_0 = [L]$.
for phase $m = 1, 2, \ldots, M$ **do**
    Set $T_m = T_{m-1} + N, q_m = 1/|A_{m-1}|, n_m = \lfloor q_m N \rfloor$.
    for $t = T_{m-1} + 1, \ldots, T_m$ **do**
        Pull $i \in A_{m-1}$ with for n_m times in order and observe $W_t(i)$.
    end for
    For all $i \in A_{m-1}$, set

$$S_m(i) = \sum_{t=T_{m-1}+1}^{T_m} W_t(i_t) \cdot \mathbb{I}\{i_t = i\}, \ \hat{w}_m(i) = \frac{S_m(i)}{n_m}.$$

    Let A_m contain the $\lceil L/2^m \rceil$ items with the highest $\hat{w}_m(i)$'s in A_{m-1}.
end for
Output the single item $i_{\text{out}} \in A_M$.

Step 1: Assume that the best arm was not eliminated prior to phase m. Then

$$\Pr(\hat{w}_m(1) < \hat{w}_m(i)) \leq \exp\left(-\frac{1}{2} n_m \Delta_i^2\right) \quad \forall i \in S_m \setminus \{1\}.$$

Step 2: The probability that the best arm is eliminated in phase m is at most

$$3 \exp\left(-\frac{T}{8 \log_2 L} \cdot \frac{\Delta_{i_m}^2}{i_m}\right)$$

where $i_m = L/2^{m+2}$.

Step 3: The failure probability can be bounded as follows:

$$\begin{aligned} 3 \sum_{m=1}^{\log_2 L} \exp\left(-\frac{T}{8 \log_2 L} \cdot \frac{\Delta_{i_m}^2}{i_m}\right) &\leq 3 \sum_{m=1}^{\log_2 L} \exp\left(-\frac{T}{8 \log_2 L} \cdot \frac{1}{\max_i i \Delta_i^{-2}}\right) \\ &= O\left(\log_2 L \exp\left(-\frac{T}{8 H_2 \log_2 L}\right)\right) \end{aligned}$$

when the hardness is measured by

$$H_2 = \max_{i \in [L]} \frac{i}{\Delta_i^2}.$$

BAI: fixed-budget

Algorithm/Instance	Reference	Failure probability e_T
$\text{UCB-E}\left(\frac{25(T-L)}{36H_1}\right)$	Audibert and Bubeck (2010)	$2TL\exp\left(-\frac{T-L}{18H_1}\right)$
SR	Audibert and Bubeck (2010)	$L(L-1)\exp\left(-\frac{T-L}{(1/2+\sum_{i=2}^{L}1/i)H_2}\right)$
$\text{UGapEb}\left(\frac{T-L}{16H_2}\right)$	Gabillon et al. (2012)	$2TL\exp\left(-\frac{T-L}{8H_2}\right)$
SAR	Bubeck et al. (2013)	$2L^2\exp\left(-\frac{T-L}{8(1/2+\sum_{i=2}^{L}1/i)H_2}\right)$
SH	Karnin et al. (2013)	$3\log_2 L\cdot\exp\left(-\frac{T}{8H_1\log_2 L}\right)$
$\text{NSE}(p)$	Shahrampour et al. (2017)	$(L-1)\exp\left(-\frac{2(T-L)}{H_p' C_p}\right)$
Stochastic Bandits	Carpentier and Locatelli (2016)	$\frac{1}{6}\exp\left(-\frac{400T}{H_2\log L}\right)$ (Lower Bound)

Shahrampour et al. (2017): $H_p' := \max_{i\neq 1}\frac{i^p}{\Delta_i^2}, \quad C_p := 2^{-p}+\sum_{i=2}^{L} i^{-p}\ \forall p>0.$

Algorithm/Instance	Reference	Failure probability e_T
$\text{UCB-E}\left(\frac{25(T-L)}{36H_1}\right)$	Audibert and Bubeck (2010)	$2TL\exp\left(-\frac{T-L}{18H_1}\right)$
SH	Karnin et al. (2013)	$3\log_2 L\cdot\exp\left(-\frac{T}{8H_1\log_2 L}\right)$
$\text{NSE}(p)$	Shahrampour et al. (2017)	$(L-1)\exp\left(-\frac{2(T-L)}{H_p' C_p}\right)$
Stochastic Bandits	Carpentier and Locatelli (2016)	$\frac{1}{6}\exp\left(-\frac{400T}{H_2\log L}\right)$ (Lower Bound)

$$H_2 := \sum_{i=1} \frac{1}{\Delta_{i^2}}, \quad H_1 := \max_{i\neq 1} \frac{i}{\Delta_i^2}, \quad H_p' := \max_{i\neq 1} \frac{i^p}{\Delta_i^2}, \quad C_p := 2^{-p} + \sum_{i=2}^{L} i^{-p}\ \forall p > 0.$$

BAI: fixed-budget

Algorithm/Instance	Reference	Failure probability e_T
$\text{UCB-E}\left(\frac{25(T-L)}{36H_1}\right)$	Audibert and Bubeck (2010)	$2TL\exp\left(-\frac{T-L}{18H_1}\right)$
SH	Karnin et al. (2013)	$3\log_2 L\cdot\exp\left(-\frac{T}{8H_1\log_2 L}\right)$
$\text{NSE}(p)$	Shahrampour et al. (2017)	$(L-1)\exp\left(-\frac{2(T-L)}{H_p' C_p}\right)$
Stochastic Bandits	Carpentier and Locatelli (2016)	$\frac{1}{6}\exp\left(-\frac{400T}{H_2\log L}\right)$ (Lower Bound)

$$H_2 := \sum_{i=1} \frac{1}{\Delta_{i^2}}, \quad H_1 := \max_{i\neq 1}\frac{i}{\Delta_i^2}, \quad H_p' := \max_{i\neq 1}\frac{i^p}{\Delta_i^2}, \quad C_p := 2^{-p} + \sum_{i=2}^{L} i^{-p}\ \forall p>0.$$

- $H_2 \le H_1 \le H_2\log(2L)$ (Audibert and Bubeck, 2010)
- Whether SH or $\text{NSE}(p)$ performs better depends on the instance, and SH does not involve a tunable parameter

Stochastic bandits

- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma(i)^2$.
- $\{W_t(i)\}_{t=1}^T$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.

♠ **Question from real life:** do we always have i.i.d. data in real life?

Stochastic bandits

- Each arm $i \in [L]$ is associated with an unknown distribution $\nu(i)$, mean $w(i)$, and variance $\sigma(i)^2$.
- $\{W_t(i)\}_{t=1}^{T}$ is the i.i.d. sequence of rewards associated with arm i during the T time steps.

♠ **Question from real life:** do we always have i.i.d. data in real life?

⇒ Stochastic bandits with adversarial corruptions

▲ Propose algorithms with near-optimal performance guarantees

▼ Demonstrate (near-)optimality by designing an appropriate corruption strategy

Case 1: Biases and Contaminations in Clinical Trials

- To test the efficacy of a medicine on randomly chosen patients.

- To test the efficacy of a medicine on randomly chosen patients.
- Possible biases and errors:

- To test the efficacy of a medicine on randomly chosen patients.
- Possible biases and errors:
  - Loss-to-follow-up,

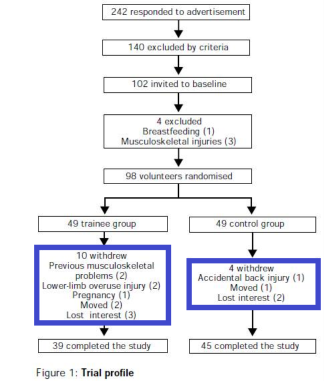


Figure 1: Loss-to-follow-up, boxed in blue.

Case 1: Biases and Contaminations in Clinical Trials

- To test the efficacy of a medicine on randomly chosen patients.
- Possible biases and errors:
  - Loss-to-follow-up,
  - Non-compliance. . .

Case 1: Biases and Contaminations in Clinical Trials

- To test the efficacy of a medicine on randomly chosen patients.
- Possible biases and errors:
  - Loss-to-follow-up,
  - Non-compliance. . .
- A lesson from COVID:
  - Not enough time to ensure i.i.d. samples!

Case 1: Biases and Contaminations in Clinical Trials

- To test the efficacy of a medicine on randomly chosen patients.
- Possible biases and errors:
  - Loss-to-follow-up,
  - Non-compliance. . .
- A lesson from COVID:
  - Not enough time to ensure i.i.d. samples!
- **How to identify the best medicine with *contaminated* data?**

Case 2: Fake Users in Online Recommendation Systems

- Paid reviews:
  - A major problem for recommender systems.

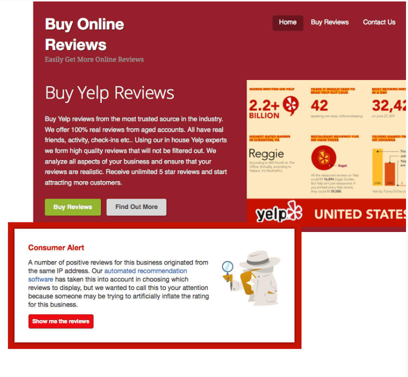


Figure 2: Buying fake reviews, and warnings about fake reviews.

Case 2: Fake Users in Online Recommendation Systems

- Paid reviews:
  - A major problem for recommender systems.
- Much effort to remove fake reviews.

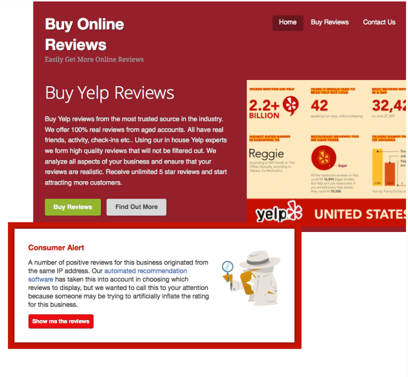


Figure 2: Buying fake reviews, and warnings about fake reviews.

Case 2: Fake Users in Online Recommendation Systems

- Paid reviews:
  - A major problem for recommender systems.
- Much effort to remove fake reviews.
- No fool-proof solution, anyone can review.

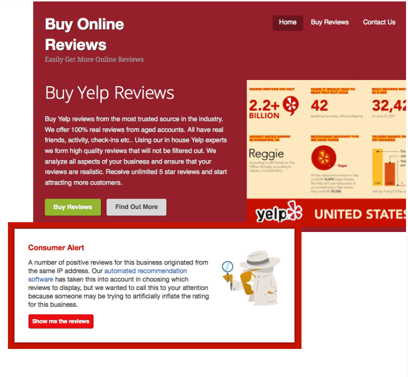


Figure 2: Buying fake reviews, and warnings about fake reviews.

Case 2: Fake Users in Online Recommendation Systems

- Paid reviews:
  - A major problem for recommender systems.
- Much effort to remove fake reviews.
- No fool-proof solution, anyone can review.
- **How to identify the best restaurants with contaminated data?**

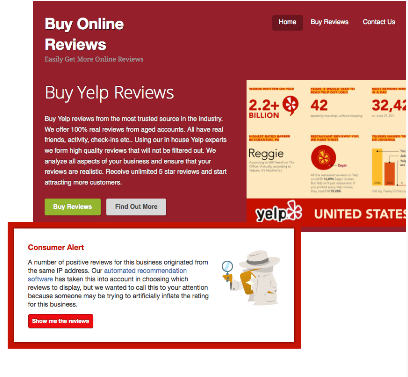


Figure 2: Buying fake reviews, and warnings about fake reviews.

Dynamic (Gupta et al., 2019)

- Ground set of L items indexed by $[L] := \{1, \ldots, L\}$.
- Each item $i \in [L]$ is associated with an unknown mean $w(i) \in (0, 1]$.

Dynamic (Gupta et al., 2019)

- Ground set of L items indexed by $[L] := \{1, \ldots, L\}$.
- Each item $i \in [L]$ is associated with an unknown mean $w(i) \in (0, 1]$.
- Amount of adversarial corruptions is bounded by the unknown corruption budget C:

$$\sum_{t=1}^{T} \max_{i \in [L]} |c_t(i)| \leq C.$$

Dynamic (Gupta et al., 2019)

- Ground set of L items indexed by $[L] := \{1, \ldots, L\}$.
- Each item $i \in [L]$ is associated with an unknown mean $w(i) \in (0, 1]$.
- Amount of adversarial corruptions is bounded by the unknown corruption budget C:

$$\sum_{t=1}^{T} \max_{i \in [L]} |c_t(i)| \leq C.$$

♠ At each time step $t = 1, \ldots, T$:

1. A stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each item i.
2. The adversary observes $\{W_t(i)\}_{i \in [L]}$, and corrupts each $W_t(i)$ with $c_t(i) \in [-1, 1]$ if the corruption budget has not been depleted:

$$\tilde{W}_t(i) = W_t(i) + c_t(i) \in [0, 1]$$

3. The agent pulls $i_t \in [L]$ and observes the corrupted reward $\tilde{W}_t(i_t)$.

Dynamic (Gupta et al., 2019)

- Ground set of L items indexed by $[L] := \{1, \ldots, L\}$.
- Each item $i \in [L]$ is associated with an unknown mean $w(i) \in (0, 1]$.
- Amount of adversarial corruptions is bounded by the unknown corruption budget C:

$$\sum_{t=1}^{T} \max_{i \in [L]} |c_t(i)| \le C.$$

♠ At each time step $t = 1, \ldots, T$:

1. A stochastic reward $W_t(i) \in [0, 1]$ is i.i.d. drawn for each item i.
2. The adversary observes $\{W_t(i)\}_{i \in [L]}$, and corrupts each $W_t(i)$ with $c_t(i) \in [-1, 1]$ if the corruption budget has not been depleted:

$$\tilde{W}_t(i) = W_t(i) + c_t(i) \in [0, 1]$$

3. The agent pulls $i_t \in [L]$ and observes the corrupted reward $\tilde{W}_t(i_t)$.

♠ At the end, the agent returns $i_{\text{out}} \in [L]$ as the recommendation.

Objective

- Assume $w(1) > w(2) \geq \ldots \geq w(L)$.
- Optimality gap of item i is $\Delta_{1,i} := w(1) - w(i)$.

Objective

- Assume $w(1) > w(2) \geq \ldots \geq w(L)$.
- Optimality gap of item i is $\Delta_{1,i} := w(1) - w(i)$.

- For fixed $\epsilon_C, \delta \in (0,1)$, an algorithm is said to be (ϵ_C, δ)-PAC (probably approximately correct) if

$$\mathbb{P}\Big[\Delta_{1,i_{\text{out}}^{\pi,T}} > \epsilon_C\Big] \leq \delta.$$

Objective

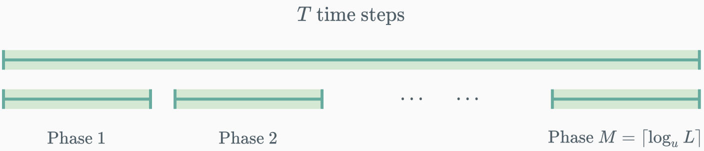
- Assume $w(1) > w(2) \geq \ldots \geq w(L)$.
- Optimality gap of item i is $\Delta_{1,i} := w(1) - w(i)$.
- For fixed $\epsilon_C, \delta \in (0,1)$, an algorithm is said to be (ϵ_C, δ)-PAC (probably approximately correct) if

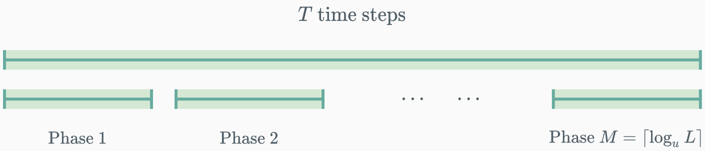
$$\mathbb{P}\left[\Delta_{1,i_{\mathrm{out}}^{\pi,T}} > \epsilon_C\right] \leq \delta.$$

♠ **Goal**: design an (ϵ_C, δ)-PAC algorithm π with both ϵ_C and δ small.

– $\epsilon_C < \Delta_{1,2}$: an (ϵ_C, δ)-PAC algorithm identifies the optimal item with probability at least $1 - \delta$.

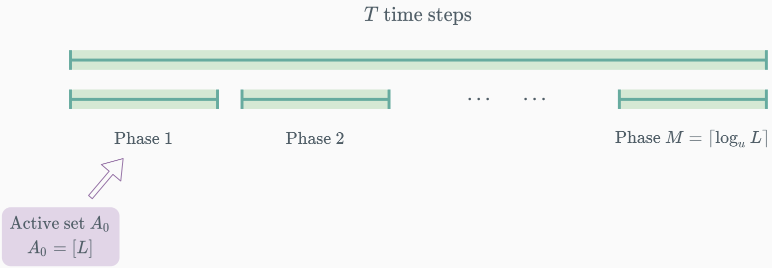
T time steps


T time steps
Phase 1
Phase 2
$\cdots$ $\cdots$
Phase $M = \lceil \log_u L \rceil$

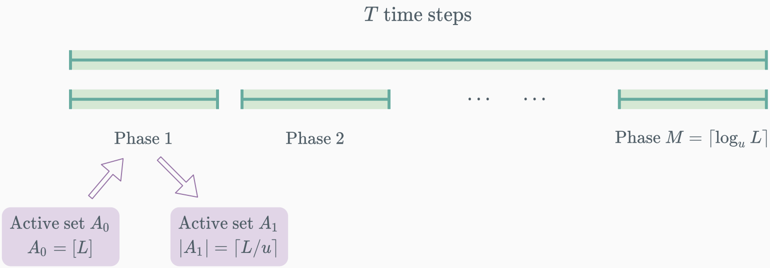


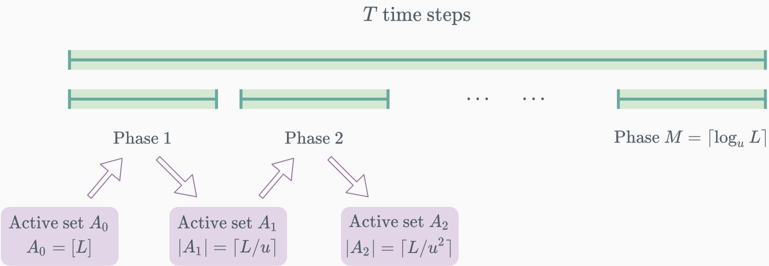
Active set A_0
$A_0 = [L]$

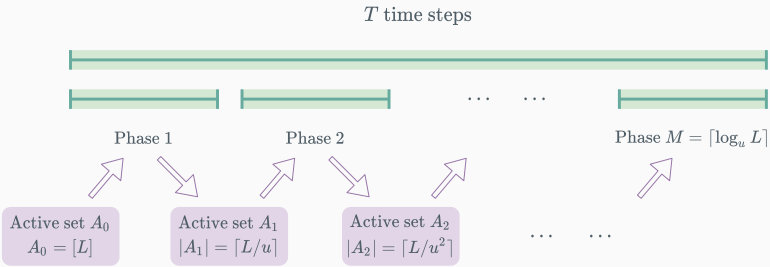
Probabilistic Sequential Shrinking Algorithm

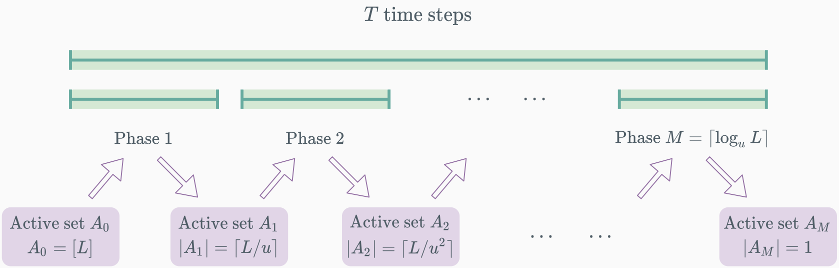


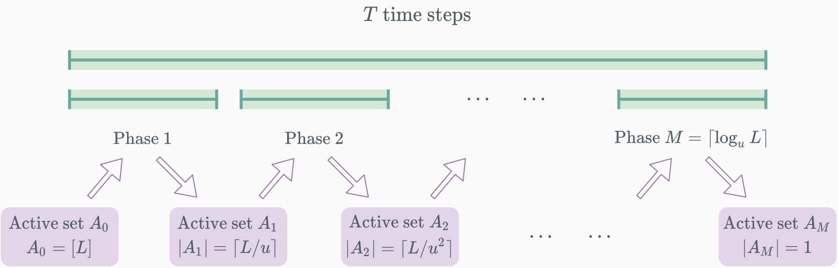
Probabilistic Sequential Shrinking Algorithm




T time steps
Phase 1
Phase 2
. . .   . . .
Phase $M = \lceil \log_u L \rceil$
Active set A_0
$A_0 = [L]$
Active set A_1
$|A_1| = \lceil L/u \rceil$
Active set A_2
$|A_2| = \lceil L/u^2 \rceil$

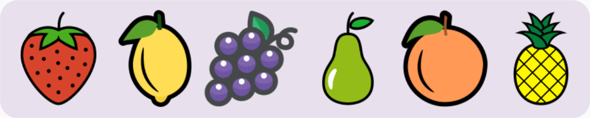

T time steps
Phase 1
Phase 2
Phase $M = \lceil \log_u L \rceil$
Active set A_0
$A_0 = [L]$
Active set A_1
$|A_1| = \lceil L/u \rceil$
Active set A_2
$|A_2| = \lceil L/u^2 \rceil$


T time steps
. . . . . .
Phase 1
Phase 2
Phase $M = \lceil \log_u L \rceil$
Active set A_0
$A_0 = [L]$
Active set A_1
$|A_1| = \lceil L/u \rceil$
Active set A_2
$|A_2| = \lceil L/u^2 \rceil$
. . . . . .
Active set A_M
$|A_M| = 1$



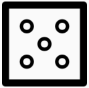
♠ How to shrink the active set?

PSS: Shrink the active set



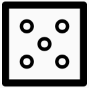
♠ Pull each active item with the same probability

♠ Pull each active item with the same probability



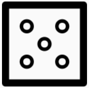
PSS: Shrink the active set

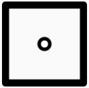
♠ Pull each active item with the same probability



$\tilde{W}_1(5) = 0.5$

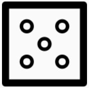
♠ Pull each active item with the same probability

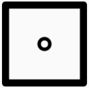




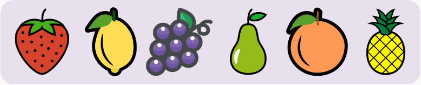
$\tilde{W}_1(5) = 0.5$

♠ Pull each active item with the same probability

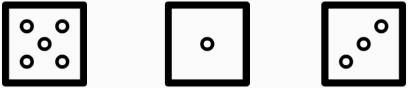




$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$

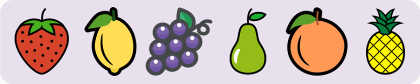


♠ Pull each active item with the same probability

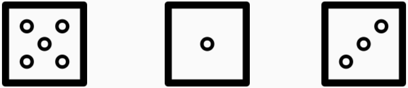


$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$

PSS: Shrink the active set

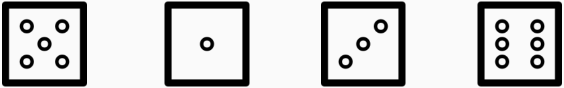


♠ Pull each active item with the same probability



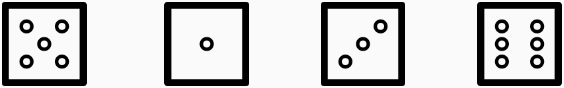
$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$

♠ Pull each active item with the same probability



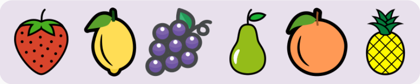
$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$

♠ Pull each active item with the same probability

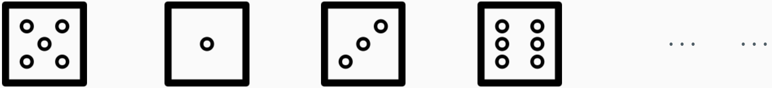


$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$ $\tilde{W}_4(6) = 0.2$

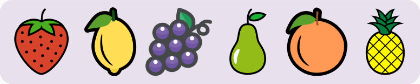
PSS: Shrink the active set



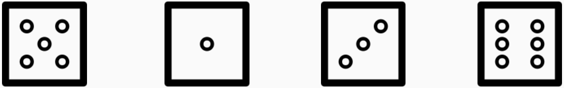
- Pull each active item with the same probability



$\tilde{W}_1(5) = 0.5$ $\quad \tilde{W}_2(1) = 0.3$ $\quad \tilde{W}_3(3) = 0.9$ $\quad \tilde{W}_4(6) = 0.2$ $\quad \cdots \quad \cdots$



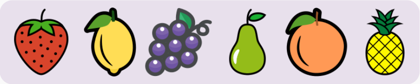
♠ Pull each active item with the same probability



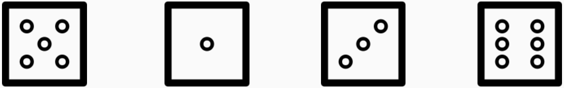
$\tilde{W}_1(5) = 0.5$ $\quad\tilde{W}_2(1) = 0.3$ $\quad\tilde{W}_3(3) = 0.9$ $\quad\tilde{W}_4(6) = 0.2$ $\quad\cdots \quad \cdots$

♠ Shrink the active set:
only keep items with high empirical means during the current phase

PSS: Shrink the active set

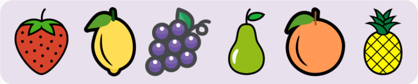


♠ Pull each active item with the same probability



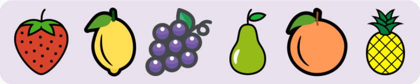
$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$ $\tilde{W}_4(6) = 0.2$ $\cdots$ $\cdots$

♠ Shrink the active set:
only keep items with high empirical means during the current phase

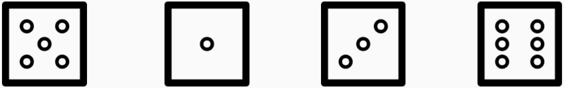


$\hat{W}_m(i)$	0.4	0.1	0.87	0.3	0.35	0.8

PSS: Shrink the active set

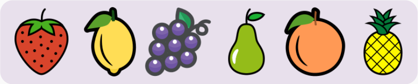


♠ Pull each active item with the same probability

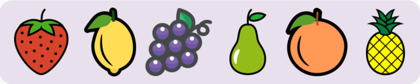


$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$ $\tilde{W}_4(6) = 0.2$ $\cdots$ $\cdots$

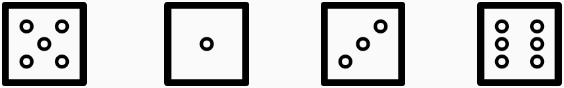
♠ Shrink the active set:
only keep items with high empirical means during the current phase



$\hat{W}_m(i)$	0.4	0.1	0.87	0.3	0.35	0.8

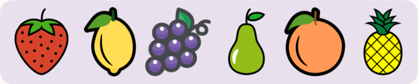


- Pull each **active** item with the **same** probability

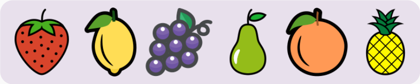


$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$ $\tilde{W}_4(6) = 0.2$ $\cdots$ $\cdots$

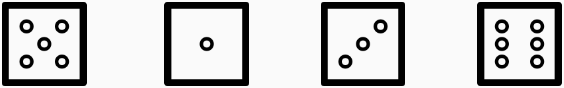
- **Shrink** the active set:
  only keep items with **high** empirical means during the **current** phase



$\hat{W}_m(i)$	**0.4**	0.1	**0.87**	0.3	0.35	**0.8**



♠ Pull each active item with the same probability



$\tilde{W}_1(5) = 0.5$ $\tilde{W}_2(1) = 0.3$ $\tilde{W}_3(3) = 0.9$ $\tilde{W}_4(6) = 0.2$ $\cdots$ $\cdots$

♠ Shrink the active set:
only keep items with high empirical means during the current phase

$\hat{W}_m(i)$	0.4	0.1	0.87	0.3	0.35	0.8

Comparison to deterministic algorithms: UP, SH

PSS(L) and Uniform Pull (UP)

- PSS(L): pulls each item for T/L times in expectation.
- UP: pulls each item for $\lfloor T/L \rfloor$ times with a deterministic schedule.

$\Rightarrow$ PSS(L): randomized version of UP.

Comparison to deterministic algorithms: UP, SH

PSS(L) and Uniform Pull (UP)

- PSS(L): pulls each item for T/L times in expectation.
- UP: pulls each item for $\lfloor T/L \rfloor$ times with a deterministic schedule.

$\Rightarrow$ PSS(L): randomized version of UP.

PSS(2) and Sequential Halving (SH) (Karnin et al., 2013)

- **Similarity**: both divide the whole horizon into $\lceil \log_2 L \rceil$ phases and halve the active set during each phase.
- **Difference**:
  - at each time step of phase m, PSS(2) chooses item $i \in A_{m-1}$ with probability $1/|A_{m-1}|$ and pulls it;
  - during phase m, SH pulls each item in A_{m-1} for exactly $\lfloor T/(\lceil \log_2 L \rceil \cdot |A_{m-1}|) \rfloor$ times according to a deterministic schedule.

$\Rightarrow$ PSS(2): randomized version of SH.

Comparison among upper bounds

NUS National University of Singapore

Comparison in stochastic bandits with adversarial corruptions

Algorithm	Order of error bound ϵ_C	Order of failure probability δ
PSS(u)	$\frac{C \log_u L}{T}$	$L(\log_u L) \exp\left[-\frac{T}{192\tilde{H}_2(w,L,u)\log_u L}\right]$

Comparison among upper bounds

Comparison in stochastic bandits with adversarial corruptions

Algorithm	Order of error bound ϵ_C	Order of failure probability δ
PSS(u)	$\dfrac{C \log_u L}{T}$	$L(\log_u L) \exp\left[-\dfrac{T}{192\tilde{H}_2(w,L,u)\log_u L}\right]$
PSS(2)	$\dfrac{C \log_2 L}{T}$	$L(\log_2 L) \exp\left[-\dfrac{T}{192\tilde{H}_2(w,L,u)\log_2 L}\right]$
SH	$\dfrac{CL \log_2 L}{T}$	$L(\log_2 L) \exp\left[-\dfrac{T}{192\tilde{H}_2(w,L,u)\log_2 L}\right]$

Comparison among upper bounds

Comparison in stochastic bandits with adversarial corruptions

Algorithm	Order of error bound ϵ_C	Order of failure probability δ
PSS(u)	$\frac{C \log_u L}{T}$	$L(\log_u L) \exp\left[-\frac{T}{192\tilde{H}_2(w,L,u)\log_u L}\right]$
PSS(2)	$\frac{C \log_2 L}{T}$	$L(\log_2 L) \exp\left[-\frac{T}{192\tilde{H}_2(w,L,u)\log_2 L}\right]$
SH	$\frac{C L \log_2 L}{T}$	$L(\log_2 L) \exp\left[-\frac{T}{192\tilde{H}_2(w,L,u)\log_2 L}\right]$
PSS(L)	$\frac{C}{T}$	$L \exp\left(-\frac{T}{192L/\Delta_{1,2}^2}\right)$
UP	$\frac{C L}{T}$	$L \exp\left(-\frac{T}{192L/\Delta_{1,2}^2}\right)$

- $\tilde{H}_2(w,L,u) = \max_{i \neq 1} \frac{\min\{u \cdot i,\ L\}}{\Delta_{1,i}^2}$: quantify difficulty of BAI.
- $H_2(w) = \max_{i \neq 1} \frac{i}{\Delta_i^2}$, $\tilde{H}_2(w,L,1) = H_2(w)$, $\tilde{H}_2(w,L,u) \le u \cdot H_2(w)$.

Corruption Strategy and Impossibility Result

Theorem 2.2

Fix $\lambda \in (0,1)$ and $\Delta \in (0,1/2)$. For any online algorithm, there is a BAI with an adversarial corruption instance over T steps, corruption budget $C = 1 + (1+\lambda)2\Delta T$, and optimality gap Δ, such that

$$\mathbb{P}[\Delta_{1,i_{\text{out}}} > 0] = \mathbb{P}[\Delta_{1,i_{\text{out}}} \geq \Delta] = \mathbb{P}[i_{\text{out}} \neq 1] \geq \frac{1}{2} \cdot \left[1 - \exp\left(-\frac{2\lambda^2 \Delta T}{3}\right)\right].$$

- $\dfrac{C}{T} > 2\Delta_{1,2}$: It is impossible for any algorithm to identify the optimal item with high probability.
- $\dfrac{C}{T} \leq \dfrac{\Delta_{1,L}}{8\lceil \log_u L \rceil}$: our work (Theorem 4.1) provides a guarantee for PSS(u).

$\Rightarrow$ The upper bound in our work (Theorem 4.1) is within a factor of $O(\log L)$ away from the largest possible upper bound on C/T in Theorem 2.2.

Summary

- Introduction on multi-armed bandit problems
- Problem formulation
    - Hardness H_1, H_2; concentration inequalities

Summary

- Introduction on multi-armed bandit problems
- Problem formulation
  - Hardness H_1, H_2; concentration inequalities
- BAI under the fixed-confidence setting
  - Algorithms: SUCCESSIVE ELIMINATION, MEDIAN ELIMINATION, TRACK & STOP
  - Lower bound: achieved by TRACK & STOP

Summary

- Introduction on multi-armed bandit problems
- Problem formulation
  - Hardness H_1, H_2; concentration inequalities
- BAI under the fixed-confidence setting
  - Algorithms: SUCCESSIVE ELIMINATION, MEDIAN ELIMINATION, TRACK & STOP
  - Lower bound: achieved by TRACK & STOP
- BAI under the fixed-budget setting
  - Algorithms: UCB-E, SEQUENTIAL HALVING
  - Gap between upper and lower bounds
  - With adversarial corruptions: PROBABILISTIC SEQUENTIAL SHRINKING

Summary

- Introduction on multi-armed bandit problems
- Problem formulation
  - Hardness H_1, H_2; concentration inequalities
- BAI under the fixed-confidence setting
  - Algorithms: SUCCESSIVE ELIMINATION, MEDIAN ELIMINATION, TRACK & STOP
  - Lower bound: achieved by TRACK & STOP
- BAI under the fixed-budget setting
  - Algorithms: UCB-E, SEQUENTIAL HALVING
  - Gap between upper and lower bounds
  - With adversarial corruptions: PROBABILISTIC SEQUENTIAL SHRINKING
- More existing works ...
  - *Multiple pure exploration*: to identify multiple arms
    CLUCB by Chen et al. (2014), EST1 and CSAR by Rejwan and Mansour (2020)
  - *Pure exploration in linear bandits*
    (Jedra and Proutiere, 2020; Yang and Tan, 2021)
  - • • •

Further exploration

- Fill the gap between upper and lower bounds for BAI under the fixed-budget setting?

Further exploration

- Fill the gap between upper and lower bounds for BAI under the fixed-budget setting?
- Identification of the arm with the highest median reward (Altschuler et al., 2019):
  More studies taking the median of rewards as the criterion are yet to be done.

Further exploration

- Fill the gap between upper and lower bounds for BAI under the fixed-budget setting?
- Identification of the arm with the highest median reward (Altschuler et al., 2019):
  More studies taking the median of rewards as the criterion are yet to be done.
- BAI in adversarial bandits (Shen, 2019; Zhong et al., 2021):
  Optimal attack strategies against regret minimization (Jun et al., 2018; Liu and Lai, 2020)
  Optimal attack strategies against pure exploration?

Thanks for listening!

https://zixinzh.github.io/homepage/conf_tutorial/

Emails: vtan@nus.edu.sg, zixin.zhong@u.nus.edu

N. Abe and P. M. Long. Associative reinforcement learning using linear probabilistic concepts. In *Proceedings of the 16th International Conference on Machine Learning*, pages 3–11, 1999.

S. Agrawal and N. Goyal. Analysis of Thompson sampling for the multi-armed bandit problem. In *Proceedings of the 25th Annual Conference on Learning Theory*, pages 39.1–39.26, 2012.

J. Altschuler, V.-E. Brunel, and A. Malek. Best arm identification for contaminated bandits. *Journal of Machine Learning Research*, 20(91):1–39, 2019.

J.-Y. Audibert and S. Bubeck. Best arm identification in multi-armed bandits. In *Proceedings of the 23th Conference on Learning Theory*, pages 41–53, 2010.

J.-Y. Audibert, S. Bubeck, et al. Minimax policies for adversarial and stochastic bandits. In *Proceedings of the 22th Conference on Learning Theory*, pages 1–122, 2009.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002a.

References II

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal of Computing*, 32(1):48–77, 2002b.

D. A. Berry, R. W. Chen, A. Zame, D. C. Heath, and L. A. Shepp. Bandit problems with infinitely many arms. *The Annals of Statistics*, pages 2103–2116, 1997.

S. Bubeck, T. Wang, and N. Viswanathan. Multiple identifications in multi-armed bandits. In *Proceedings of the 30th International Conference on Machine Learning*, pages 258–265, 2013.

A. Carpentier and A. Locatelli. Tight (lower) bounds for the fixed budget best arm identification bandit problem. In *Proceedings of the 29th Conference on Learning Theory*, pages 590–604, 2016.

A. Carpentier and M. Valko. Simple regret for infinitely many armed bandits. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1133–1141, 2015.

N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

References III

S. Chen, T. Lin, I. King, M. R. Lyu, and W. Chen. Combinatorial pure exploration of multi-armed bandits. In *Proceedings of the 27th Advances in Neural Information Processing Systems*, pages 379–387. 2014.

D. P. Dubhashi and A. Panconesi. *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press, 2009.

E. Even-Dar, S. Mannor, and Y. Mansour. PAC bounds for multi-armed bandit and markov decision processes. In *Proceedings of the 15th Annual International Conference on Computational Learning Theory*, pages 255–270, 2002.

V. Gabillon, M. Ghavamzadeh, and A. Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, 2012.

A. Garivier and E. Kaufmann. Optimal best arm identification with fixed confidence. In V. Feldman, A. Rakhlin, and O. Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 998–1027, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR.

A. Gupta, T. Koren, and K. Talwar. Better algorithms for stochastic bandits with adversarial corruptions. In *Proceedings of the 32nd Conference on Learning Theory*, pages 1562–1578, 2019.

K. Jamieson and R. Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *Proceedings of the 48th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6, 2014.

Y. Jedra and A. Proutiere. Optimal best-arm identification in linear bandits. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 10007–10017. Curran Associates, Inc., 2020.

K.-S. Jun, L. Li, Y. Ma, and J. Zhu. Adversarial attacks on stochastic bandits. In *Proceedings of the 31st Advances in Neural Information Processing Systems*, pages 3640–3649, 2018.

S. Kalyanakrishnan, A. Tewari, P. Auer, and P. Stone. Pac subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Conference on Machine Learning*, pages 655–662, 2012.

Z. Karnin, T. Koren, and O. Somekh. Almost optimal exploration in multi-armed bandits. In *Proceedings of the 13th International Conference on Machine Learning*, pages 1238–1246, 2013.

Y. Kuroki, L. Xu, A. Miyauchi, J. Honda, and M. Sugiyama. Polynomial-time algorithms for multiple-arm identification with full-bandit feedback. *Neural Computation*, 32(9):1733–1773, 2020.

B. Kveton, C. Szepesvari, Z. Wen, and A. Ashkan. Cascading bandits: Learning to rank in the cascade model. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 767–776, 2015a.

B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvári. Combinatorial cascading bandits. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 1450–1458, 2015b.

T. L. Lai. Adaptive treatment allocation and the multi-armed bandit problem. *The Annals of Statistics*, 15(3):1091 – 1114, 1987.

S. Li, B. Wang, S. Zhang, and W. Chen. Contextual combinatorial cascading bandits. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 1245–1253, 2016.

References VI

G. Liu and L. Lai. Action-manipulation attacks on stochastic bandits. In *Proceedings of the 45th International Conference on Acoustics, Speech and Signal Processing*, pages 3112–3116, 2020.

S. Mannor and J. N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5(Jun): 623–648, 2004.

I. Rejwan and Y. Mansour. Top-k combinatorial bandits with full-bandit feedback. In *Proceedings of the 31st International Conference on Algorithmic Learning Theory*, pages 752–776, 2020.

D. Russo and B. Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

S. Shahrampour, M. Noshad, and V. Tarokh. On sequential elimination algorithms for best-arm identification in multi-armed bandits. *IEEE Transactions on Signal Processing*, 65(16):4281–4292, 2017. doi: 10.1109/TSP.2017.2706192.

C. Shen. Universal best arm identification. *IEEE Transactions on Signal Processing*, 67(17):4464–4478, 2019.

J. Yang and V. Tan. Towards minimax optimal best arm identification in linear bandits. *arXiv preprint arXiv:2105.13017*, 2021.

References VII

Z. Zhong, W. C. Cheung, and V. Tan. Probabilistic sequential shrinking: A best arm identification algorithm for stochastic bandits with corruptions. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

J. Zimmert and Y. Seldin. Tsallis-INF: An optimal algorithm for stochastic and adversarial bandits. *Journal of Machine Learning Research*, 22:28–1, 2021.