



# INTRODUCTION TO MULTI-ARMED BANDITS AND REINFORCEMENT LEARNING

Training School on Machine Learning for Communications  
Paris, 23-25 September 2019

# Who am I ?

Hi, I'm Lilian Besson

- ▶ finishing my PhD in telecommunication and machine learning
- ▶ under supervision of Prof. Christophe Moy at IETR & CentraleSupélec in Rennes (France)
- ▶ and Dr. Émilie Kaufmann in Inria in Lille

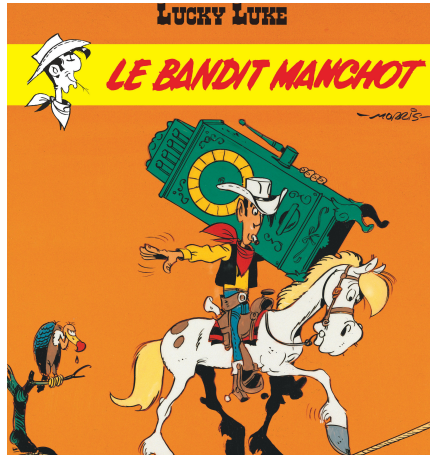
*Thanks to Émilie Kaufmann for most of the slides material!*

- ▶ [Lilian.Besson @ Inria.fr](mailto:Lilian.Besson@Inria.fr)
- ▶ ↪ [perso.crans.org/besson/](https://perso.crans.org/besson/) & [GitHub.com/Naareen](https://GitHub.com/Naareen)



# What is a *bandit*?

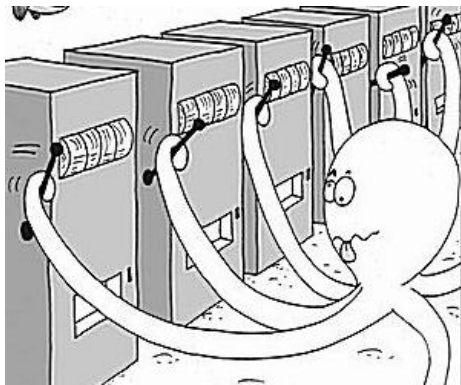
It's an old name for a casino machine!



→ © Dargaud, [Lucky Luke tome 18](#).

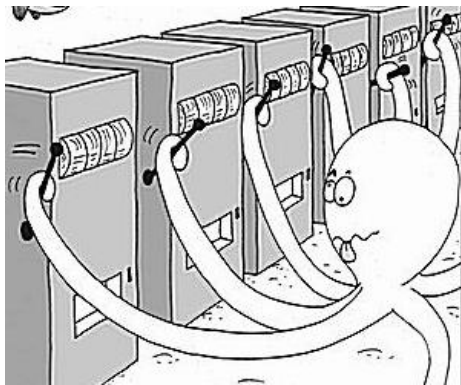
# WHY BANDITS?

# Make money in a casino?



A (single) **agent** facing (multiple) **arms** in a Multi-Armed Bandit.

# Make money in a casino?



A (single) **agent** facing (multiple) **arms** in a Multi-Armed Bandit.

# NO!

## Clinical trials

- ▶  $K$  treatments for a given symptom (with unknown effect)



- ▶ What treatment should be allocated to the next patient, based on responses observed on previous patients?

# Sequential resource allocation

## Clinical trials

- ▶  $K$  treatments for a given symptom (with unknown effect)



- ▶ What treatment should be allocated to the next patient, based on responses observed on previous patients?

## Online advertisement

- ▶  $K$  adds that can be displayed

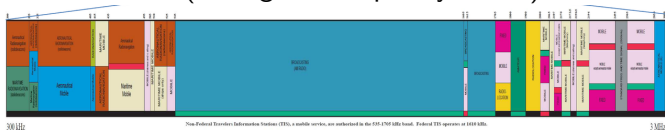


- ▶ Which add should be displayed for a user, based on the previous clicks of previous (similar) users?



## Opportunistic Spectrum Access

- ▶  $K$  radio channels (orthogonal frequency bands)

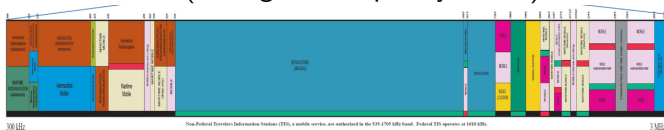


- ▶ In which channel should a radio device send a packet, based on the quality of its previous communications?

# Dynamic channel selection

## Opportunistic Spectrum Access

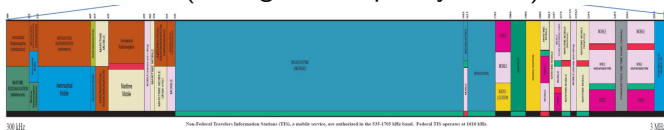
- ▶  $K$  radio channels (orthogonal frequency bands)



- ▶ In which channel should a radio device send a packet, based on the quality of its previous communications? ↪ [see the next talk at 4pm !](#)

## Opportunistic Spectrum Access

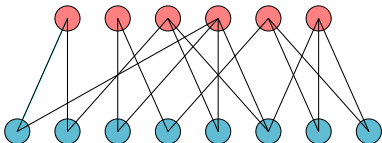
- ▶  $K$  radio channels (orthogonal frequency bands)



- ▶ In which channel should a radio device send a packet, based on the quality of its previous communications? ↪ [see the next talk at 4pm !](#)

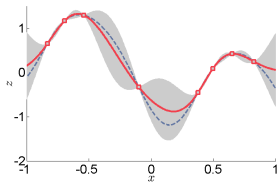
## Communications in presence of a central controller

- ▶  $K$  assignments from  $n$  users to  $m$  antennas ( $\rightsquigarrow$  *combinatorial* bandit)



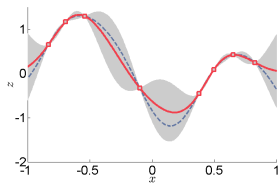
- ▶ How to select the next *matching* based on the throughput observed in previous communications?

## Numerical experiments (bandits for “black-box” optimization)



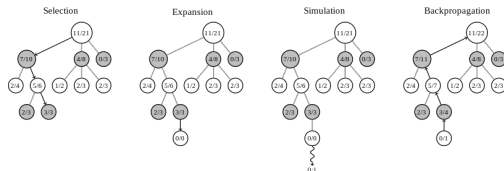
- ▶ where to evaluate a costly function in order to find its maximum?

## Numerical experiments (bandits for “black-box” optimization)



- ▶ where to evaluate a costly function in order to find its maximum?

## Artificial intelligence for games



- ▶ where to choose the next evaluation to perform in order to find the best move to play next?

# Why talking about bandits today?

- ▶ rewards maximization in a stochastic bandit model  
= **the simplest Reinforcement Learning (RL) problem** (one state)  
⇒ good introduction to RL !
- ▶ bandits showcase the important **exploration/exploitation dilemma**
- ▶ **bandit tools** are useful for RL  
(UCRL, bandit-based MCTS for planning in games. . .)
- ▶ a **rich literature** to tackle many specific applications
- ▶ bandits have application **beyond RL** (i.e. without “reward”)
- ▶ and bandits have great applications to Cognitive Radio  
↔ **see the next talk at 4pm !**

# Outline of this talk

- ▶ Multi-armed Bandit
- ▶ Performance measure (regret) and first strategies
- ▶ Best possible regret? Lower bounds
- ▶ Mixing Exploration and Exploitation
- ▶ The Optimism Principle and Upper Confidence Bounds (UCB) Algorithms
- ▶ A Bayesian Look at the Multi-Armed Bandit Model
- ▶ Many extensions of the stationary single-player bandit models
- ▶ Summary

# The Multi-Armed Bandit Setup

$K$  arms  $\Leftrightarrow K$  rewards streams  $(X_{a,t})_{t \in \mathbb{N}}$



At round  $t$ , an agent:

- ▶ chooses an arm  $A_t$
- ▶ receives a reward  $R_t = X_{A_t,t}$  (from the environment)

**Sequential** sampling strategy (**bandit algorithm**):

$$A_{t+1} = F_t(A_1, R_1, \dots, A_t, R_t).$$

**Goal:** Maximize sum of rewards  $\sum_{t=1}^T R_t$ .



# The Stochastic Multi-Armed Bandit Setup

$K$  arms  $\Leftrightarrow K$  probability distributions :  $\nu_a$  has mean  $\mu_a$



$\nu_1$



$\nu_2$



$\nu_3$



$\nu_4$



$\nu_5$

At round  $t$ , an agent:

- ▶ chooses an arm  $A_t$
- ▶ receives a reward  $R_t = X_{A_t,t} \sim \nu_{A_t}$  (i.i.d. from a distribution)

Sequential sampling strategy (**bandit algorithm**):

$$A_{t+1} = F_t(A_1, R_1, \dots, A_t, R_t).$$

**Goal:** Maximize sum of rewards  $\mathbb{E} \left[ \sum_{t=1}^T R_t \right]$ .

# Discover bandits by playing this online demo!

Total Total  
Reward Plays  
14 24



1

	Arm 1	Arm 2	Arm 3	Arm 4	Arm 5
<b>Rewards:</b>	6	2	2	2	2
<b>Pulls:</b>	8	4	4	4	4
<b>Estimated Probs:</b>	0.750	0.500	0.500	0.500	0.500
<b>UCBs:</b>	1.641	1.761	1.761	1.761	1.761

↪ Interactive demo on this web-page

[perso.crans.org/besson/phd/MAB\\_interactive\\_demo/](http://perso.crans.org/besson/phd/MAB_interactive_demo/)

## Historical motivation [Thompson 1933]

 $B(\mu_1)$  $B(\mu_2)$  $B(\mu_3)$  $B(\mu_4)$  $B(\mu_5)$ 

For the  $t$ -th patient in a clinical study,

- ▶ chooses a **treatment**  $A_t$
- ▶ observes a (Bernoulli) **response**  $R_t \in \{0, 1\} : \mathbb{P}(R_t = 1 | A_t = a) = \mu_a$

**Goal:** maximize the expected number of patients healed.

**Modern motivation** (\$\$\$\$) [Li et al, 2010]  
(recommender systems, online advertisement, etc)



$\nu_1$



$\nu_2$



$\nu_3$



$\nu_4$



$\nu_5$

For the  $t$ -th visitor of a website,

- ▶ recommend a **movie**  $A_t$
- ▶ observe a **rating**  $R_t \sim \nu_{A_t}$  (e.g.  $R_t \in \{1, \dots, 5\}$ )

**Goal:** maximize the sum of ratings.

## Opportunistic spectrum access [Zhao et al. 10] [Anandkumar et al. 11]

*streams indicating channel quality*

Channel 1	$X_{1,1}$	$X_{1,2}$	...	$X_{1,t}$	...	$X_{1,T}$	$\sim \nu_1$
Channel 2	$X_{2,1}$	$X_{2,2}$	...	$X_{2,t}$	...	$X_{2,T}$	$\sim \nu_2$
...	...	...	...	...	...	...	...
Channel $K$	$X_{K,1}$	$X_{K,2}$	...	$X_{K,t}$	...	$X_{K,T}$	$\sim \nu_K$

At round  $t$ , the device:

- ▶ selects a channel  $A_t$
- ▶ observes the quality of its communication  $R_t = X_{A_t,t} \in [0, 1]$

**Goal:** Maximize the overall quality of communications.

↪ see the next talk at 4pm !

# PERFORMANCE MEASURE AND FIRST STRATEGIES

# Regret of a bandit algorithm

**Bandit instance:**  $\nu = (\nu_1, \nu_2, \dots, \nu_K)$ , mean of arm  $a$ :  $\mu_a = \mathbb{E}_{X \sim \nu_a}[X]$ .

$$\mu_\star = \max_{a \in \{1, \dots, K\}} \mu_a \quad \text{and} \quad a_\star = \operatorname{argmax}_{a \in \{1, \dots, K\}} \mu_a.$$

Maximizing rewards  $\Leftrightarrow$  selecting  $a_\star$  as much as possible  
 $\Leftrightarrow$  minimizing the **regret** [Robbins, 52]

$$\mathcal{R}_\nu(\mathcal{A}, T) := \underbrace{T\mu_\star}_{\text{sum of rewards of an oracle strategy always selecting } a_\star} - \underbrace{\mathbb{E} \left[ \sum_{t=1}^T R_t \right]}_{\text{sum of rewards of the strategy } \mathcal{A}}$$

# Regret of a bandit algorithm

**Bandit instance:**  $\nu = (\nu_1, \nu_2, \dots, \nu_K)$ , mean of arm  $a$ :  $\mu_a = \mathbb{E}_{X \sim \nu_a}[X]$ .

$$\mu_\star = \max_{a \in \{1, \dots, K\}} \mu_a \quad \text{and} \quad a_\star = \operatorname{argmax}_{a \in \{1, \dots, K\}} \mu_a.$$

Maximizing rewards  $\Leftrightarrow$  selecting  $a_\star$  as much as possible  
 $\Leftrightarrow$  minimizing the **regret** [Robbins, 52]

$$\mathcal{R}_\nu(\mathcal{A}, T) := \underbrace{T\mu_\star}_{\substack{\text{sum of rewards of} \\ \text{an oracle strategy} \\ \text{always selecting } a_\star}} - \underbrace{\mathbb{E} \left[ \sum_{t=1}^T R_t \right]}_{\substack{\text{sum of rewards of} \\ \text{the strategy } \mathcal{A}}}$$

What regret rate can we achieve?

$\Rightarrow$  consistency:  $\mathcal{R}_\nu(\mathcal{A}, T)/T \Rightarrow 0$  (when  $T \rightarrow \infty$ )

$\Rightarrow$  can we be more precise?



# Regret decomposition

$N_a(t)$  : number of selections of arm  $a$  in the first  $t$  rounds

$\Delta_a := \mu_\star - \mu_a$  : sub-optimality gap of arm  $a$

## Regret decomposition

$$\mathcal{R}_\nu(\mathcal{A}, T) = \sum_{a=1}^K \Delta_a \mathbb{E}[N_a(T)].$$

# Regret decomposition

$N_a(t)$  : number of selections of arm  $a$  in the first  $t$  rounds

$\Delta_a := \mu_\star - \mu_a$  : sub-optimality gap of arm  $a$

## Regret decomposition

$$\mathcal{R}_\nu(\mathcal{A}, T) = \sum_{a=1}^K \Delta_a \mathbb{E}[N_a(T)].$$

**Proof.**

$$\begin{aligned} \mathcal{R}_\nu(\mathcal{A}, T) &= \mu_\star T - \mathbb{E}\left[\sum_{t=1}^T X_{A_t, t}\right] = \mu_\star T - \mathbb{E}\left[\sum_{t=1}^T \mu_{A_t}\right] \\ &= \mathbb{E}\left[\sum_{t=1}^T (\mu_\star - \mu_{A_t})\right] \\ &= \sum_{a=1}^K \underbrace{(\mu_\star - \mu_a)}_{\Delta_a} \mathbb{E}\left[\underbrace{\sum_{t=1}^T \mathbb{1}(A_t = a)}_{N_a(T)}\right]. \end{aligned}$$

# Regret decomposition

$N_a(t)$  : number of selections of arm  $a$  in the first  $t$  rounds

$\Delta_a := \mu_\star - \mu_a$  : sub-optimality gap of arm  $a$

## Regret decomposition

$$\mathcal{R}_\nu(\mathcal{A}, T) = \sum_{a=1}^K \Delta_a \mathbb{E}[N_a(T)].$$

A strategy with small regret should:

- ▶ select not too often arms for which  $\Delta_a > 0$  (sub-optimal arms)
- ▶ ... which requires to try all arms to estimate the values of the  $\Delta_a$

⇒ Exploration / Exploitation trade-off !

# Two naive strategies

► **Idea 1 :**

⇒ EXPLORATION

Draw each arm  $T/K$  times

$$\hookrightarrow \mathcal{R}_\nu(\mathcal{A}, T) = \left( \frac{1}{K} \sum_{a: \mu_a > \mu_*} \Delta_a \right) T = \Omega(T)$$

# Two naive strategies

► **Idea 1 :**

⇒ EXPLORATION

Draw each arm  $T/K$  times

$$\hookrightarrow \mathcal{R}_\nu(\mathcal{A}, T) = \left( \frac{1}{K} \sum_{a: \mu_a > \mu_*} \Delta_a \right) T = \Omega(T)$$

► **Idea 2 :** Always trust the empirical best arm

⇒ EXPLOITATION

$A_{t+1} = \operatorname{argmax}_{a \in \{1, \dots, K\}} \hat{\mu}_a(t)$  using estimates of the unknown means  $\mu_a$

$$\hat{\mu}_a(t) = \frac{1}{N_a(t)} \sum_{s=1}^t X_{a,s} \mathbb{1}_{(A_s=a)}$$

$$\hookrightarrow \mathcal{R}_\nu(\mathcal{A}, T) \geq (1 - \mu_1) \times \mu_2 \times (\mu_1 - \mu_2) T = \Omega(T)$$

(with  $K = 2$  Bernoulli arms of means  $\mu_1 \neq \mu_2$ )

## A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ EXPLORATION followed by EXPLOITATION

# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ **EXPLORATION** followed by **EXPLOITATION**

Analysis for  $K = 2$  arms. If  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

$$\begin{aligned} \mathcal{R}_\nu(\text{ETC}, T) &= \Delta \mathbb{E}[N_2(T)] \\ &= \Delta \mathbb{E}[m + (T - Km) \mathbb{1}(\hat{a} = 2)] \\ &\leq \Delta m + (\Delta T) \times \mathbb{P}(\hat{\mu}_{2,m} \geq \hat{\mu}_{1,m}) \end{aligned}$$

$\hat{\mu}_{a,m}$ : empirical mean of the first  $m$  observations from arm  $a$

# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ EXPLORATION followed by EXPLOITATION

Analysis for  $K = 2$  arms. If  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

$$\begin{aligned} \mathcal{R}_\nu(\text{ETC}, T) &= \Delta \mathbb{E}[N_2(T)] \\ &= \Delta \mathbb{E}[m + (T - Km) \mathbb{1}(\hat{a} = 2)] \\ &\leq \Delta m + (\Delta T) \times \mathbb{P}(\hat{\mu}_{2,m} \geq \hat{\mu}_{1,m}) \end{aligned}$$

$\hat{\mu}_{a,m}$ : empirical mean of the first  $m$  observations from arm  $a$

⇒ requires a concentration inequality



# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ EXPLORATION followed by EXPLOITATION

Analysis for two arms.  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

**Assumption 1:**  $\nu_1, \nu_2$  are bounded in  $[0, 1]$ .

$$\begin{aligned}\mathcal{R}_\nu(T) &= \Delta \mathbb{E}[N_2(T)] \\ &= \Delta \mathbb{E}[m + (T - Km)\mathbb{1}(\hat{a} = 2)] \\ &\leq \Delta m + (\Delta T) \times \exp(-m\Delta^2/2)\end{aligned}$$

$\hat{\mu}_{a,m}$ : empirical mean of the first  $m$  observations from arm  $a$

⇒ Hoeffding's inequality

# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ EXPLORATION followed by EXPLOITATION

Analysis for two arms.  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

**Assumption 2:**  $\nu_1 = \mathcal{N}(\mu_1, \sigma^2)$ ,  $\nu_2 = \mathcal{N}(\mu_2, \sigma^2)$  are **Gaussian arms**.

$$\begin{aligned} \mathcal{R}_\nu(\text{ETC}, T) &= \Delta \mathbb{E}[N_2(T)] \\ &= \Delta \mathbb{E}[m + (T - Km)\mathbb{1}(\hat{a} = 2)] \\ &\leq \Delta m + (\Delta T) \times \exp(-m\Delta^2/4\sigma^2) \end{aligned}$$

$\hat{\mu}_{a,m}$ : empirical mean of the first  $m$  observations from arm  $a$

⇒ **Gaussian tail inequality**

# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ EXPLORATION followed by EXPLOITATION

Analysis for two arms.  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

**Assumption 2:**  $\nu_1 = \mathcal{N}(\mu_1, \sigma^2)$ ,  $\nu_2 = \mathcal{N}(\mu_2, \sigma^2)$  are **Gaussian arms**.

$$\begin{aligned} \mathcal{R}_\nu(\text{ETC}, T) &= \Delta \mathbb{E}[N_2(T)] \\ &= \Delta \mathbb{E}[m + (T - Km)\mathbb{1}(\hat{a} = 2)] \\ &\leq \Delta m + (\Delta T) \times \exp(-m\Delta^2/4\sigma^2) \end{aligned}$$

$\hat{\mu}_{a,m}$ : empirical mean of the first  $m$  observations from arm  $a$

⇒ **Gaussian tail inequality**

# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ **EXPLORATION** followed by **EXPLOITATION**

Analysis for two arms.  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

**Assumption:**  $\nu_1 = \mathcal{N}(\mu_1, \sigma^2)$ ,  $\nu_2 = \mathcal{N}(\mu_2, \sigma^2)$  are **Gaussian arms**.

For  $m = \frac{4\sigma^2}{\Delta^2} \log\left(\frac{T\Delta^2}{4\sigma^2}\right)$ ,

$$\mathcal{R}_\nu(\text{ETC}, T) \leq \frac{4\sigma^2}{\Delta} \left[ \log\left(\frac{T\Delta^2}{2}\right) + 1 \right] = \mathcal{O}\left(\frac{1}{\Delta} \log(T)\right).$$

# A better idea: Explore-Then-Commit (ETC)

Given  $m \in \{1, \dots, T/K\}$ ,

- ▶ draw each arm  $m$  times
- ▶ compute the empirical best arm  $\hat{a} = \operatorname{argmax}_a \hat{\mu}_a(Km)$
- ▶ keep playing this arm until round  $T$

$$A_{t+1} = \hat{a} \text{ for } t \geq Km$$

⇒ EXPLORATION followed by EXPLOITATION

Analysis for two arms.  $\mu_1 > \mu_2$ ,  $\Delta := \mu_1 - \mu_2$ .

**Assumption:**  $\nu_1 = \mathcal{N}(\mu_1, \sigma^2)$ ,  $\nu_2 = \mathcal{N}(\mu_2, \sigma^2)$  are **Gaussian arms**.

For  $m = \frac{4\sigma^2}{\Delta^2} \log\left(\frac{T\Delta^2}{4\sigma^2}\right)$ ,

$$\mathcal{R}_\nu(\text{ETC}, T) \leq \frac{4\sigma^2}{\Delta} \left[ \log\left(\frac{T\Delta^2}{2}\right) + 1 \right] = \mathcal{O}\left(\frac{1}{\Delta} \log(T)\right).$$

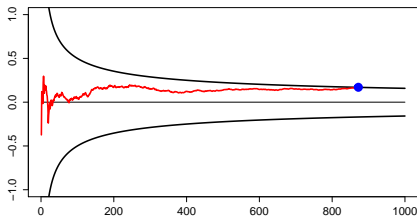
+ logarithmic regret!

– requires the knowledge of  $T$  ( $\simeq$  OKAY) and  $\Delta$  (NOT OKAY)

# Sequential Explore-Then-Commit (2 Gaussian arms)

- ▶ explore uniformly until the random time

$$\tau = \inf \left\{ t \in \mathbb{N} : |\hat{\mu}_1(t) - \hat{\mu}_2(t)| > \sqrt{\frac{8\sigma^2 \log(T/t)}{t}} \right\}$$

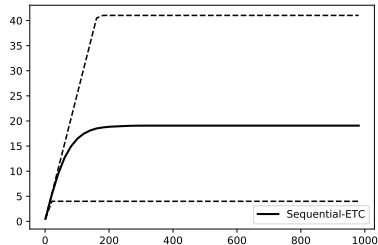
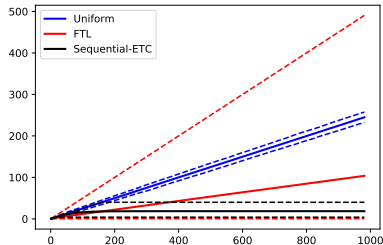


- ▶  $\hat{a}_\tau = \operatorname{argmax}_a \hat{\mu}_a(\tau)$  and  $(A_{t+1} = \hat{a}_\tau)$  for  $t \in \{\tau + 1, \dots, T\}$

$$\mathcal{R}_\nu(\text{S-ETC}, T) \leq \frac{4\sigma^2}{\Delta} \log(T\Delta^2) + C\sqrt{\log(T)} = \mathcal{O}\left(\frac{1}{\Delta} \log(T)\right).$$

→ same regret rate, without knowing  $\Delta$  [Garivier et al. 2016]

Two Gaussian arms:  $\nu_1 = \mathcal{N}(1, 1)$  and  $\nu_2 = \mathcal{N}(1.5, 1)$



Expected regret estimated over  $N = 500$  runs for Sequential-ETC versus our two naive baselines.

(dashed lines: empirical 0.05% and 0.95% quantiles of the regret)

# Is this a good regret rate?

For two-armed Gaussian bandits,

$$\mathcal{R}_\nu(\text{ETC}, T) \lesssim \frac{4\sigma^2}{\Delta} \log(T\Delta^2) = \mathcal{O}\left(\frac{1}{\Delta} \log(T)\right).$$

⇒ **problem-dependent logarithmic** regret bound  
 $\mathcal{R}_\nu(\text{algo}, T) = \mathcal{O}(\log(T)).$

**Observation:** blows up when  $\Delta$  tends to zero...

$$\begin{aligned} \mathcal{R}_\nu(\text{ETC}, T) &\lesssim \min \left[ \frac{4\sigma^2}{\Delta} \log(T\Delta^2), \Delta T \right] \\ &\leq \sqrt{T} \min_{u>0} \left[ \frac{4\sigma^2}{u} \log(u^2), u \right] \leq C\sqrt{T}. \end{aligned}$$

⇒ **problem-independent square-root** regret bound  
 $\mathcal{R}_\nu(\text{algo}, T) = \mathcal{O}(\sqrt{T}).$



# BEST POSSIBLE REGRET? LOWER BOUNDS

# The Lai and Robbins lower bound

**Context:** a **parametric bandit model** where each arm is parameterized by its mean  $\nu = (\nu_{\mu_1}, \dots, \nu_{\mu_K})$ ,  $\mu_a \in \mathcal{I}$ .

distributions  $\nu \Leftrightarrow \mu = (\mu_1, \dots, \mu_K)$  means

**Key tool:** **Kullback-Leibler divergence.**

## Kullback-Leibler divergence

$$\text{kl}(\mu, \mu') := \text{KL}(\nu_\mu, \nu_{\mu'}) = \mathbb{E}_{X \sim \nu_\mu} \left[ \log \frac{d\nu_\mu}{d\nu_{\mu'}}(X) \right]$$

## Theorem [Lai and Robbins, 1985]

For uniformly efficient algorithms ( $\mathcal{R}_\mu(\mathcal{A}, T) = o(T^\alpha)$  for all  $\alpha \in (0, 1)$  and  $\mu \in \mathcal{I}^K$ ),

$$\mu_a < \mu_\star \implies \liminf_{T \rightarrow \infty} \frac{\mathbb{E}_\mu[N_a(T)]}{\log T} \geq \frac{1}{\text{kl}(\mu_a, \mu_\star)}.$$

# The Lai and Robbins lower bound

**Context:** a **parametric bandit model** where each arm is parameterized by its mean  $\nu = (\nu_{\mu_1}, \dots, \nu_{\mu_K})$ ,  $\mu_a \in \mathcal{I}$ .

distributions  $\nu \Leftrightarrow \mu = (\mu_1, \dots, \mu_K)$  means

**Key tool:** **Kullback-Leibler divergence.**

## Kullback-Leibler divergence

$$\text{kl}(\mu, \mu') := \frac{(\mu - \mu')^2}{2\sigma^2} \quad (\text{Gaussian bandits with variance } \sigma^2)$$

## Theorem [Lai and Robbins, 1985]

For uniformly efficient algorithms ( $\mathcal{R}_\mu(\mathcal{A}, T) = o(T^\alpha)$  for all  $\alpha \in (0, 1)$  and  $\mu \in \mathcal{I}^K$ ),

$$\mu_a < \mu_* \implies \liminf_{T \rightarrow \infty} \frac{\mathbb{E}_\mu[N_a(T)]}{\log T} \geq \frac{1}{\text{kl}(\mu_a, \mu_*)}.$$

# The Lai and Robbins lower bound

**Context:** a **parametric bandit model** where each arm is parameterized by its mean  $\nu = (\nu_{\mu_1}, \dots, \nu_{\mu_K})$ ,  $\mu_a \in \mathcal{I}$ .

distributions  $\nu \Leftrightarrow \mu = (\mu_1, \dots, \mu_K)$  means

**Key tool:** **Kullback-Leibler divergence.**

## Kullback-Leibler divergence

$$\text{kl}(\mu, \mu') := \mu \log \left( \frac{\mu}{\mu'} \right) + (1 - \mu) \log \left( \frac{1 - \mu}{1 - \mu'} \right) \quad (\text{Bernoulli bandits})$$

## Theorem [Lai and Robbins, 1985]

For uniformly efficient algorithms ( $\mathcal{R}_\mu(\mathcal{A}, T) = o(T^\alpha)$  for all  $\alpha \in (0, 1)$  and  $\mu \in \mathcal{I}^K$ ),

$$\mu_a < \mu_* \implies \liminf_{T \rightarrow \infty} \frac{\mathbb{E}_\mu[N_a(T)]}{\log T} \geq \frac{1}{\text{kl}(\mu_a, \mu_*)}.$$

# Some room for better algorithms?

- ▶ For two-armed Gaussian bandits, ETC satisfies

$$\mathcal{R}_\nu(\text{ETC}, T) \lesssim \frac{4\sigma^2}{\Delta} \log(T\Delta^2) = \mathcal{O}\left(\frac{1}{\Delta} \log(T)\right),$$

with  $\Delta = |\mu_1 - \mu_2|$ .

- ▶ The Lai and Robbins' lower bound yields, for large values of  $T$ ,

$$\mathcal{R}_\nu(\mathcal{A}, T) \gtrsim \frac{2\sigma^2}{\Delta} \log(T\Delta^2) = \Omega\left(\frac{1}{\Delta} \log(T)\right),$$

as  $\text{kl}(\mu_1, \mu_2) = \frac{(\mu_1 - \mu_2)^2}{2\sigma^2}$ .

⇒ Explore-Then-Commit is **not asymptotically optimal**.

# MIXING EXPLORATION AND EXPLOITATION

## A simple strategy: $\varepsilon$ -greedy

The  $\varepsilon$ -greedy rule [Sutton and Barton, 98] is the simplest way to alternate exploration and exploitation.

### $\varepsilon$ -greedy strategy

At round  $t$ ,

- ▶ with probability  $\varepsilon$

$$A_t \sim \mathcal{U}(\{1, \dots, K\})$$

- ▶ with probability  $1 - \varepsilon$

$$A_t = \operatorname{argmax}_{a=1, \dots, K} \hat{\mu}_a(t).$$

$\Rightarrow$  Linear regret:  $\mathcal{R}_\nu(\varepsilon\text{-greedy}, T) \geq \varepsilon \frac{K-1}{K} \Delta_{\min} T.$

$$\Delta_{\min} = \min_{a: \mu_a < \mu_*} \Delta_a.$$

# A simple strategy: $\varepsilon$ -greedy

A simple fix: make  $\varepsilon$  decreasing!

## $\varepsilon_t$ -greedy strategy

At round  $t$ ,

- ▶ with probability  $\varepsilon_t := \min\left(1, \frac{K}{d^2 t}\right)$  probability  $\searrow$  with  $t$

$$A_t \sim \mathcal{U}(\{1, \dots, K\})$$

- ▶ with probability  $1 - \varepsilon_t$

$$A_t = \operatorname{argmax}_{a=1, \dots, K} \hat{\mu}_a(t-1).$$

## Theorem [Auer et al. 02]

If  $0 < d \leq \Delta_{\min}$ ,  $\mathcal{R}_\nu(\varepsilon_t\text{-greedy}, T) = \mathcal{O}\left(\frac{1}{d^2} K \log(T)\right)$ .

$\implies$  requires the knowledge of a lower bound on  $\Delta_{\min}$ .



# THE OPTIMISM PRINCIPLE

## UPPER CONFIDENCE BOUNDS ALGORITHMS

# The optimism principle

**Step 1:** construct a set of statistically plausible models

- ▶ For each arm  $a$ , build a confidence interval  $\mathcal{I}_a(t)$  on the mean  $\mu_a$  :

$$\mathcal{I}_a(t) = [\text{LCB}_a(t), \text{UCB}_a(t)]$$

LCB = Lower Confidence Bound

UCB = Upper Confidence Bound

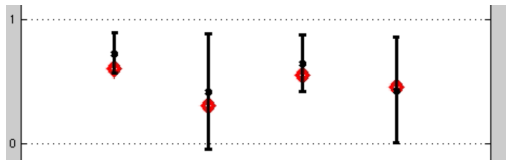


Figure: Confidence intervals on the means after  $t$  rounds

# The optimism principle

**Step 2:** act as if the best possible model were the true model  
(*“optimism in face of uncertainty”*)

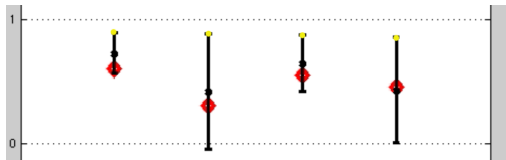


Figure: Confidence intervals on the means after  $t$  rounds

$$\text{Optimistic bandit model} = \underset{\mu \in \mathcal{C}(t)}{\operatorname{argmax}} \max_{a=1, \dots, K} \mu_a$$

► That is, select

$$A_{t+1} = \underset{a=1, \dots, K}{\operatorname{argmax}} \operatorname{UCB}_a(t).$$

# Optimistic Algorithms

## Building Confidence Intervals

### Analysis of $UCB(\alpha)$

# How to build confidence intervals?

We need  $\text{UCB}_a(t)$  such that

$$\mathbb{P}(\mu_a \leq \text{UCB}_a(t)) \gtrsim 1 - 1/t.$$

$\implies$  tool: concentration inequalities

**Example:** rewards are  $\sigma^2$  sub-Gaussian

$$\mathbb{E}[Z] = \mu \quad \text{and} \quad \mathbb{E}\left[e^{\lambda(Z-\mu)}\right] \leq e^{\lambda^2\sigma^2/2}. \quad (1)$$

## Hoeffding inequality

$Z_i$  i.i.d. satisfying (1). For all (**fixed**)  $s \geq 1$

$$\mathbb{P}\left(\frac{Z_1 + \dots + Z_s}{s} \geq \mu + x\right) \leq e^{-sx^2/(2\sigma^2)}$$

- ▶  $\nu_a$  bounded in  $[0, 1]$ :  $1/4$  sub-Gaussian
- ▶  $\nu_a = \mathcal{N}(\mu_a, \sigma^2)$ :  $\sigma^2$  sub-Gaussian

# How to build confidence intervals?

We need  $UCB_a(t)$  such that

$$\mathbb{P}(\mu_a \leq UCB_a(t)) \gtrsim 1 - 1/t.$$

$\implies$  tool: concentration inequalities

**Example:** rewards are  $\sigma^2$  sub-Gaussian

$$\mathbb{E}[Z] = \mu \quad \text{and} \quad \mathbb{E}\left[e^{\lambda(Z-\mu)}\right] \leq e^{\lambda^2\sigma^2/2}. \quad (1)$$

## Hoeffding inequality

$Z_i$  i.i.d. satisfying (1). For all (**fixed**)  $s \geq 1$

$$\mathbb{P}\left(\frac{Z_1 + \dots + Z_s}{s} \leq \mu - x\right) \leq e^{-sx^2/(2\sigma^2)}$$

- ▶  $\nu_a$  bounded in  $[0, 1]$ :  $1/4$  sub-Gaussian
- ▶  $\nu_a = \mathcal{N}(\mu_a, \sigma^2)$ :  $\sigma^2$  sub-Gaussian

# How to build confidence intervals?

We need  $UCB_a(t)$  such that

$$\mathbb{P}(\mu_a \leq UCB_a(t)) \gtrsim 1 - 1/t.$$

$\implies$  tool: concentration inequalities


**Example:** rewards are  $\sigma^2$  sub-Gaussian

$$\mathbb{E}[Z] = \mu \quad \text{and} \quad \mathbb{E}\left[e^{\lambda(Z-\mu)}\right] \leq e^{\lambda^2\sigma^2/2}. \quad (1)$$

## Hoeffding inequality

$Z_i$  i.i.d. satisfying (1). For all (**fixed**)  $s \geq 1$

$$\mathbb{P}\left(\frac{Z_1 + \dots + Z_s}{s} \leq \mu - x\right) \leq e^{-sx^2/(2\sigma^2)}$$

 Cannot be used directly in a bandit model as the number of observations  $s$  from each arm is random!

# How to build confidence intervals?

- ▶  $N_a(t) = \sum_{s=1}^t \mathbb{1}_{(A_s=a)}$  number of selections of  $a$  after  $t$  rounds
- ▶  $\hat{\mu}_{a,s} = \frac{1}{s} \sum_{k=1}^s Y_{a,k}$  average of the first  $s$  observations from arm  $a$
- ▶  $\hat{\mu}_a(t) = \hat{\mu}_{a,N_a(t)}$  empirical estimate of  $\mu_a$  after  $t$  rounds

## Hoeffding inequality + union bound

$$\mathbb{P} \left( \mu_a \leq \hat{\mu}_a(t) + \sigma \sqrt{\frac{\alpha \log(t)}{N_a(t)}} \right) \geq 1 - \frac{1}{t^{\frac{\alpha}{2}-1}}$$



# How to build confidence intervals?

- ▶  $N_a(t) = \sum_{s=1}^t \mathbb{1}_{(A_s=a)}$  number of selections of  $a$  after  $t$  rounds
- ▶  $\hat{\mu}_{a,s} = \frac{1}{s} \sum_{k=1}^s Y_{a,k}$  average of the first  $s$  observations from arm  $a$
- ▶  $\hat{\mu}_a(t) = \hat{\mu}_{a,N_a(t)}$  empirical estimate of  $\mu_a$  after  $t$  rounds

## Hoeffding inequality + union bound

$$\mathbb{P} \left( \mu_a \leq \hat{\mu}_a(t) + \sigma \sqrt{\frac{\alpha \log(t)}{N_a(t)}} \right) \geq 1 - \frac{1}{t^{\frac{\alpha}{2}-1}}$$

### Proof.

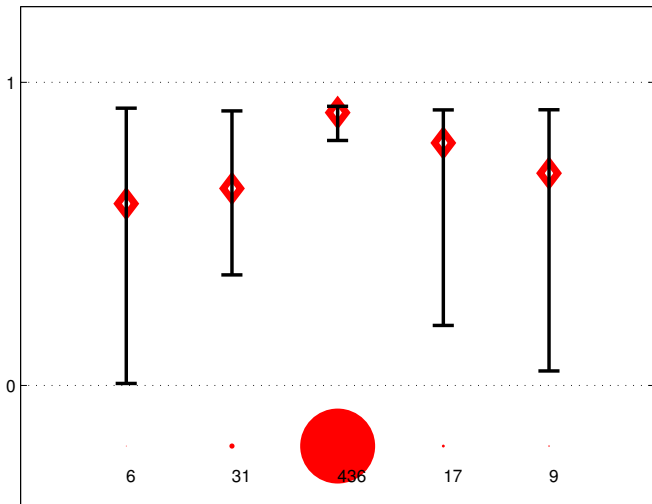
$$\begin{aligned} \mathbb{P} \left( \mu_a > \hat{\mu}_a(t) + \sigma \sqrt{\frac{\alpha \log(t)}{N_a(t)}} \right) &\leq \mathbb{P} \left( \exists s \leq t : \mu_a > \hat{\mu}_{a,s} + \sigma \sqrt{\frac{\alpha \log(t)}{s}} \right) \\ &\leq \sum_{s=1}^t \mathbb{P} \left( \hat{\mu}_{a,s} < \mu_a - \sigma \sqrt{\frac{\alpha \log(t)}{s}} \right) \leq \sum_{s=1}^t \frac{1}{t^{\alpha/2}} = \frac{1}{t^{\alpha/2-1}}. \end{aligned}$$

# A first UCB algorithm

UCB( $\alpha$ ) selects  $A_{t+1} = \operatorname{argmax}_a \text{UCB}_a(t)$  where

$$\text{UCB}_a(t) = \underbrace{\hat{\mu}_a(t)}_{\text{exploitation term}} + \underbrace{\sqrt{\frac{\alpha \log(t)}{N_a(t)}}}_{\text{exploration bonus}} .$$

- ▶ this form of UCB was first proposed for Gaussian rewards [Katehakis and Robbins, 95]
- ▶ popularized by [Auer et al. 02] for bounded rewards:  
UCB1, for  $\alpha = 2$   $\hookrightarrow$  see the next talk at 4pm !
- ▶ the analysis was UCB( $\alpha$ ) was further refined to hold for  $\alpha > 1/2$  in that case [Bubeck, 11, Cappé et al. 13]



# Optimistic Algorithms

Building Confidence Intervals

**Analysis of UCB( $\alpha$ )**

## Theorem [Auer et al, 02]

UCB( $\alpha$ ) with parameter  $\alpha = 2$  satisfies

$$\mathcal{R}_\nu(\text{UCB1}, T) \leq 8 \left( \sum_{a: \mu_a < \mu_\star} \frac{1}{\Delta_a} \right) \log(T) + \left( 1 + \frac{\pi^2}{3} \right) \left( \sum_{a=1}^K \Delta_a \right).$$

## Theorem

For every  $\alpha > 1$  and every sub-optimal arm  $a$ , there exists a constant

$$C_\alpha > 0 \text{ such that } \mathbb{E}_\mu[N_a(T)] \leq \frac{4\alpha}{(\mu_\star - \mu_a)^2} \log(T) + C_\alpha.$$

It follows that

$$\mathcal{R}_\nu(\text{UCB}(\alpha), T) \leq 4\alpha \left( \sum_{a: \mu_a < \mu_\star} \frac{1}{\Delta_a} \right) \log(T) + KC_\alpha.$$

- ▶ Several ways to solve the exploration/exploitation trade-off
  - ▶ Explore-Then-Commit
  - ▶  $\epsilon$ -greedy
  - ▶ Upper Confidence Bound algorithms
- ▶ Good concentration inequalities are crucial to build good UCB algorithms!
- ▶ Performance lower bounds motivate the design of (optimal) algorithms

# A BAYESIAN LOOK AT THE MAB MODEL

# Bayesian Bandits

Two points of view

Bayes-UCB

Thompson Sampling



1952 Robbins, formulation of the MAB problem

1985 Lai and Robbins: lower bound, first asymptotically optimal algorithm

1987 Lai, asymptotic regret of kl-UCB

1995 Agrawal, UCB algorithms

1995 Katehakis and Robbins, a UCB algorithm for Gaussian bandits

2002 Auer et al: UCB1 with finite-time regret bound

2009 UCB-V, MOSS...

2011,13 Cappé et al: finite-time regret bound for kl-UCB

# Historical perspective

- 1933 Thompson: a Bayesian mechanism for clinical trials
- 1952 Robbins, formulation of the MAB problem
- 1956 Bradt et al, Bellman: optimal solution of a Bayesian MAB problem
- 1979 Gittins: first Bayesian index policy
- 1985 Lai and Robbins: lower bound, first asymptotically optimal algorithm
- 1985 Berry and Fristedt: Bandit Problems, a survey on the Bayesian MAB
- 1987 Lai, asymptotic regret of  $\text{kl-UCB}$  + study of its Bayesian regret
- 1995 Agrawal, UCB algorithms
- 1995 Katehakis and Robbins, a UCB algorithm for Gaussian bandits
- 2002 Auer et al: UCB1 with finite-time regret bound
- 2009 UCB-V, MOSS...
- 2010 Thompson Sampling is re-discovered
- 2011,13 Cappé et al: finite-time regret bound for  $\text{kl-UCB}$
- 2012,13 Thompson Sampling is asymptotically optimal

# Frequentist versus Bayesian bandit

$$\nu_{\mu} = (\nu^{\mu_1}, \dots, \nu^{\mu_K}) \in (\mathcal{P})^K.$$

- ▶ Two probabilistic models

two points of view!

Frequentist model	Bayesian model
$\mu_1, \dots, \mu_K$ unknown parameters	$\mu_1, \dots, \mu_K$ drawn from a prior distribution : $\mu_a \sim \pi_a$
arm $a$ : $(Y_{a,s})_s \stackrel{\text{i.i.d.}}{\sim} \nu^{\mu_a}$	arm $a$ : $(Y_{a,s})_s   \mu \stackrel{\text{i.i.d.}}{\sim} \nu^{\mu_a}$

- ▶ The regret can be computed in each case

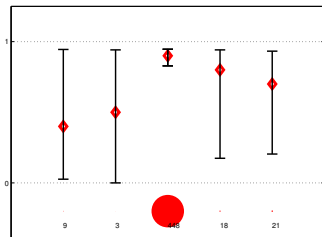
Frequentist Regret (regret)	Bayesian regret (Bayes risk)
$\mathcal{R}_{\mu}(\mathcal{A}, T) = \mathbb{E}_{\mu} \left[ \sum_{t=1}^T (\mu_{\star} - \mu_{A_t}) \right]$	$\begin{aligned} \mathcal{R}^{\pi}(\mathcal{A}, T) &= \mathbb{E}_{\mu \sim \pi} \left[ \sum_{t=1}^T (\mu_{\star} - \mu_{A_t}) \right] \\ &= \int \mathcal{R}_{\mu}(\mathcal{A}, T) d\pi(\mu) \end{aligned}$

# Frequentist and Bayesian algorithms

- ▶ Two types of tools to build bandit algorithms:

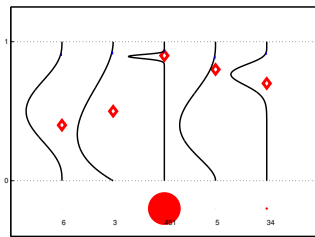
## Frequentist tools

MLE estimators of the means  
Confidence Intervals



## Bayesian tools

Posterior distributions  
 $\pi_a^t = \mathcal{L}(\mu_a | Y_{a,1}, \dots, Y_{a,N_a(t)})$



# Example: Bernoulli bandits

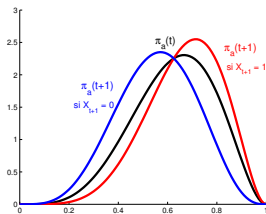
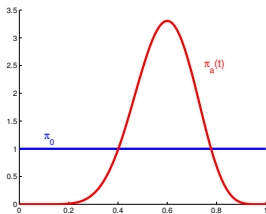
Bernoulli bandit model  $\mu = (\mu_1, \dots, \mu_K)$

► **Bayesian view:**  $\mu_1, \dots, \mu_K$  are random variables

prior distribution :  $\mu_a \sim \mathcal{U}([0, 1])$

⇒ posterior distribution:

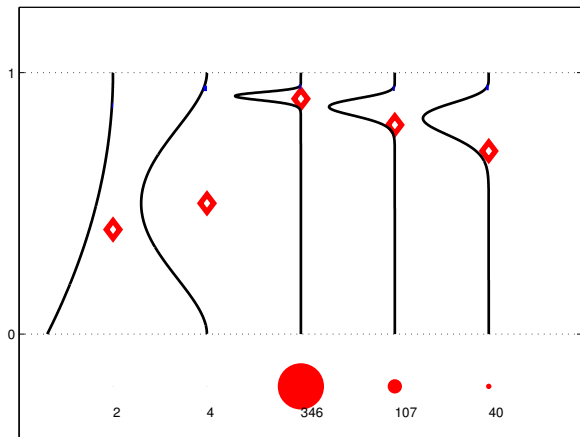
$$\begin{aligned}\pi_a(t) &= \mathcal{L}(\mu_a | R_1, \dots, R_t) \\ &= \text{Beta}(\underbrace{S_a(t) + 1}_{\text{\#ones}}, \underbrace{N_a(t) - S_a(t) + 1}_{\text{\#zeros}})\end{aligned}$$



$S_a(t) = \sum_{s=1}^t R_s \mathbb{1}_{(A_s=a)}$  sum of the rewards from arm  $a$

# Bayesian algorithm

A **Bayesian bandit algorithm** exploits the posterior distributions of the means to decide which arm to select.



# Bayesian Bandits

Two points of view

**Bayes-UCB**

Thompson Sampling

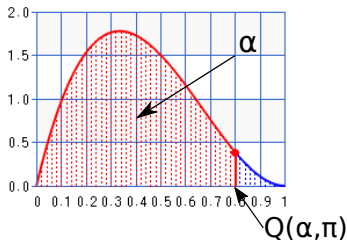
# The Bayes-UCB algorithm

- ▶  $\Pi_0 = (\pi_1(0), \dots, \pi_K(0))$  be a prior distribution over  $(\mu_1, \dots, \mu_K)$
- ▶  $\Pi_t = (\pi_1(t), \dots, \pi_K(t))$  be the posterior distribution over the means  $(\mu_1, \dots, \mu_K)$  after  $t$  observations

The **Bayes-UCB algorithm** chooses at time  $t$

$$A_{t+1} = \operatorname{argmax}_{a=1, \dots, K} Q\left(1 - \frac{1}{t(\log t)^c}, \pi_a(t)\right)$$

where  $Q(\alpha, \pi)$  is the quantile of order  $\alpha$  of the distribution  $\pi$ .





# The Bayes-UCB algorithm

- ▶  $\Pi_0 = (\pi_1(0), \dots, \pi_K(0))$  be a prior distribution over  $(\mu_1, \dots, \mu_K)$
- ▶  $\Pi_t = (\pi_1(t), \dots, \pi_K(t))$  be the posterior distribution over the means  $(\mu_1, \dots, \mu_K)$  after  $t$  observations

The **Bayes-UCB algorithm** chooses at time  $t$

$$A_{t+1} = \operatorname{argmax}_{a=1, \dots, K} Q\left(1 - \frac{1}{t(\log t)^c}, \pi_a(t)\right)$$

where  $Q(\alpha, \pi)$  is the quantile of order  $\alpha$  of the distribution  $\pi$ .

**Bernoulli reward with uniform prior:**

- ▶  $\pi_a(0) \stackrel{i.i.d.}{\sim} \mathcal{U}([0, 1]) = \text{Beta}(1, 1)$
- ▶  $\pi_a(t) = \text{Beta}(S_a(t) + 1, N_a(t) - S_a(t) + 1)$

# The Bayes-UCB algorithm

- ▶  $\Pi_0 = (\pi_1(0), \dots, \pi_K(0))$  be a prior distribution over  $(\mu_1, \dots, \mu_K)$
- ▶  $\Pi_t = (\pi_1(t), \dots, \pi_K(t))$  be the posterior distribution over the means  $(\mu_1, \dots, \mu_K)$  after  $t$  observations

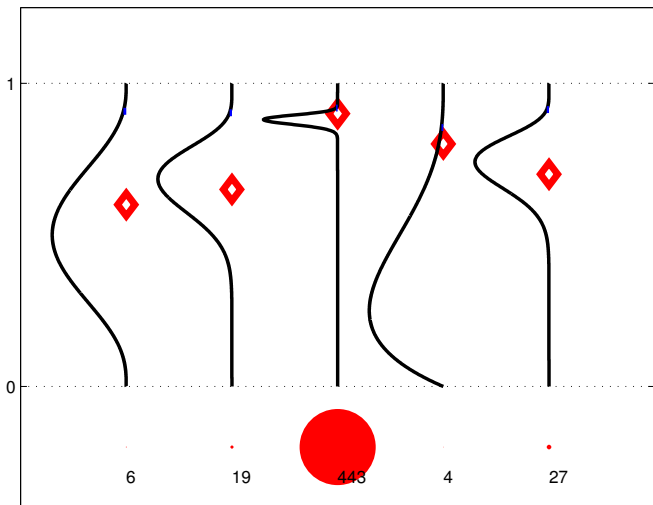
The **Bayes-UCB algorithm** chooses at time  $t$

$$A_{t+1} = \operatorname{argmax}_{a=1, \dots, K} Q \left( 1 - \frac{1}{t(\log t)^c}, \pi_a(t) \right)$$

where  $Q(\alpha, \pi)$  is the quantile of order  $\alpha$  of the distribution  $\pi$ .

Gaussian rewards with Gaussian prior:

- ▶  $\pi_a(0) \stackrel{i.i.d}{\sim} \mathcal{N}(0, \kappa^2)$
- ▶  $\pi_a(t) = \mathcal{N} \left( \frac{S_a(t)}{N_a(t) + \sigma^2/\kappa^2}, \frac{\sigma^2}{N_a(t) + \sigma^2/\kappa^2} \right)$



- ▶ Bayes-UCB is **asymptotically optimal** for Bernoulli rewards

Theorem [K., Cappé, Garivier 2012]

Let  $\varepsilon > 0$ . The Bayes-UCB algorithm using a uniform prior over the arms and parameter  $c \geq 5$  satisfies

$$\mathbb{E}_{\mu}[N_a(T)] \leq \frac{1 + \varepsilon}{\text{kl}(\mu_a, \mu_{\star})} \log(T) + o_{\varepsilon, c}(\log(T)).$$

# Bayesian Bandits

Insights from the Optimal Solution

Bayes-UCB

Thompson Sampling

# Historical perspective

- 1933 Thompson: in the context of clinical trial, the allocation of a treatment should be some increasing function of its **posterior probability to be optimal**
- 2010 Thompson Sampling rediscovered under different names
  - Bayesian Learning Automaton [Granmo, 2010]
  - Randomized probability matching [Scott, 2010]
- 2011 An empirical evaluation of Thompson Sampling: **an efficient algorithm**, beyond simple bandit models
  - [Li and Chapelle, 2011]
- 2012 First (logarithmic) **regret bound** for Thompson Sampling
  - [Agrawal and Goyal, 2012]
- 2012 Thompson Sampling is **asymptotically optimal for Bernoulli bandits**
  - [K., Korda and Munos, 2012][Agrawal and Goyal, 2013]
- 2013- Many **successful uses of Thompson Sampling** beyond Bernoulli bandits (contextual bandits, reinforcement learning)

# Thompson Sampling

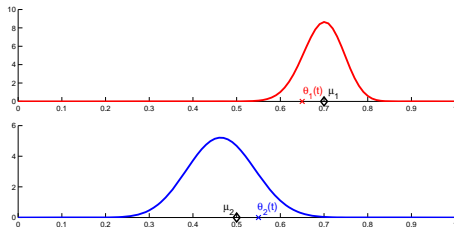
## Two equivalent interpretations:

- ▶ “select an arm at random according to its probability of being the best”
- ▶ “draw a possible bandit model from the posterior distribution and act optimally in this sampled model”

≠ optimistic

## Thompson Sampling: a randomized Bayesian algorithm

$$\begin{cases} \forall a \in \{1..K\}, \theta_a(t) \sim \pi_a(t) \\ A_{t+1} = \operatorname{argmax}_{a=1..K} \theta_a(t). \end{cases}$$



# Thompson Sampling is asymptotically optimal

## Problem-dependent regret

$$\forall \varepsilon > 0, \mathbb{E}_{\mu}[N_a(T)] \leq \frac{1 + \varepsilon}{\text{kl}(\mu_a, \mu_{\star})} \log(T) + o_{\mu, \varepsilon}(\log(T)).$$

This result holds:

- ▶ for **Bernoulli bandits**, with a **uniform prior** [K. Korda, Munos 12][Agrawal and Goyal 13]
- ▶ for **Gaussian bandits**, with **Gaussian prior** [Agrawal and Goyal 17]
- ▶ for **exponential family bandits**, with **Jeffrey's prior** [Korda et al. 13]

## Problem-independent regret [Agrawal and Goyal 13]

For Bernoulli and Gaussian bandits, Thompson Sampling satisfies

$$\mathcal{R}_{\mu}(\text{TS}, T) = O\left(\sqrt{KT \log(T)}\right).$$

- ▶ Thompson Sampling is also **asymptotically optimal for Gaussian with unknown mean and variance** [Honda and Takemura, 14]



# Understanding Thompson Sampling

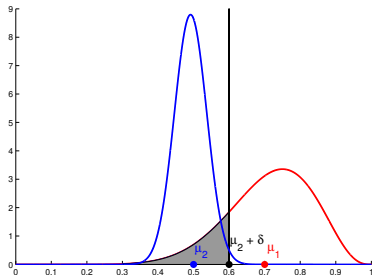
- ▶ a key ingredient in the analysis of [K. Korda and Munos 12]

## Proposition

There exists constants  $b = b(\mu) \in (0, 1)$  and  $C_b < \infty$  such that

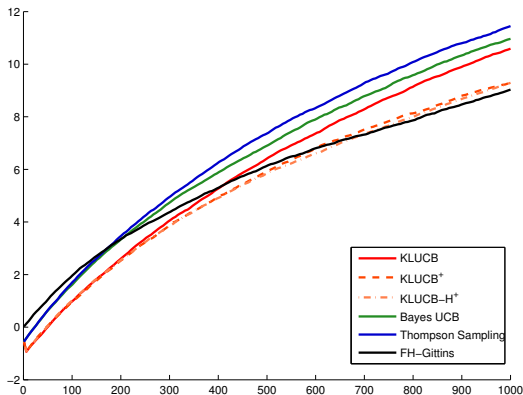
$$\sum_{t=1}^{\infty} \mathbb{P} \left( N_1(t) \leq t^b \right) \leq C_b.$$

$\{N_1(t) \leq t^b\} = \{ \text{there exists a time range of length at least } t^{1-b} - 1$   
with no draw of arm 1 } }



# Bayesian versus Frequentist algorithms

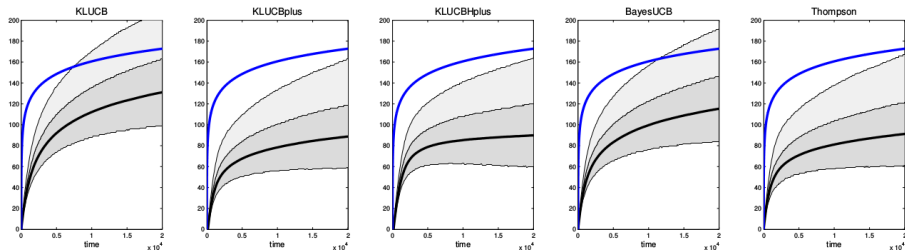
- ▶ Short horizon,  $T = 1000$  (average over  $N = 10000$  runs)



$K = 2$  Bernoulli arms  $\mu_1 = 0.2, \mu_2 = 0.25$

# Bayesian versus Frequentist algorithms

- ▶ Long horizon,  $T = 20000$  (average over  $N = 50000$  runs)



$K = 10$  Bernoulli arms bandit problem

$$\mu = [0.1 \ 0.05 \ 0.05 \ 0.05 \ 0.02 \ 0.02 \ 0.02 \ 0.01 \ 0.01 \ 0.01]$$

# OTHER BANDIT MODELS

# Other Bandit Models

**Many different extensions**

Piece-wise stationary bandits

Multi-player bandits

Most famous extensions:

- ▶ (centralized) multiple-actions

↔ Implemented in our library **SMPyBandits!**

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)

↔ Implemented in our library **SMPyBandits!**

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
  - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)

↔ Implemented in our library **SMPyBandits!**



# Many other bandits models and problems (1/2)

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
  - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)
- ▶ non stationary

↔ Implemented in our library **SMPyBandits!**

# Many other bandits models and problems (1/2)

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
  - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)
- ▶ non stationary
  - ▶ **piece-wise stationary / abruptly changing**

↔ Implemented in our library **SMPyBandits!**

# Many other bandits models and problems (1/2)

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
  - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)
- ▶ non stationary
  - ▶ **piece-wise stationary / abruptly changing**
  - ▶ **slowly-varying**

↔ Implemented in our library **SMPyBandits!**

# Many other bandits models and problems (1/2)

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
  - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)
- ▶ non stationary
  - ▶ **piece-wise stationary / abruptly changing**
  - ▶ slowly-varying
  - ▶ adversarial...

↔ Implemented in our library **SMPyBandits!**

Most famous extensions:

- ▶ (centralized) multiple-actions
  - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
  - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)
- ▶ non stationary
  - ▶ **piece-wise stationary / abruptly changing**
  - ▶ slowly-varying
  - ▶ adversarial...
- ▶ (decentralized) collaborative/communicating bandits over a graph

↔ Implemented in our library **SMPyBandits!**

Most famous extensions:

- ▶ (centralized) multiple-actions
    - ▶ **multiple choice** : choose  $m \in \{2, \dots, K - 1\}$  arms (fixed size)
    - ▶ **combinatorial** : choose a subset of arms  $S \subset \{1, \dots, K\}$  (large space)
  - ▶ non stationary
    - ▶ **piece-wise stationary / abruptly changing**
    - ▶ slowly-varying
    - ▶ adversarial...
  - ▶ (decentralized) collaborative/communicating bandits over a graph
  - ▶ (decentralized) non communicating multi-player bandits
- ↔ Implemented in our library **SMPyBandits!**

# Many other bandits models and problems (2/2)

And many more extensions. . .

- ▶ non stochastic, Markov models rested/restless
- ▶ best arm identification (vs reward maximization)
  - ▶ fixed budget setting
  - ▶ fixed confidence setting
  - ▶ PAC (probably approximately correct) algorithms
- ▶ bandits with (differential) privacy constraints
- ▶ for some applications (content recommendation)
  - ▶ contextual bandits : observe a reward and a *context* ( $C_t \in \mathbb{R}^d$ )
  - ▶ cascading bandits
  - ▶ delayed feedback bandits
- ▶ structured bandits (low-rank, many-armed, Lipschitz etc)
- ▶  $\mathcal{X}$ -armed, continuous-armed bandits

# Other Bandit Models

Many different extensions

**Piece-wise stationary bandits**

Multi-player bandits



## Stationary MAB problems

Arm  $a$  gives rewards sampled from **the same distribution** for any time step

$$\forall t, r_a(t) \stackrel{\text{iid}}{\sim} \nu_a = \mathcal{B}(\mu_a).$$

# Piece-wise stationary bandits

## Stationary MAB problems

Arm  $a$  gives rewards sampled from **the same distribution** for any time step

$$\forall t, r_a(t) \stackrel{\text{iid}}{\sim} \nu_a = \mathcal{B}(\mu_a).$$

## Non stationary MAB problems?

(possibly) **different distributions** for any time step !

$$\forall t, r_a(t) \stackrel{\text{iid}}{\sim} \nu_a(t) = \mathcal{B}(\mu_a(t)).$$

$\Rightarrow$  harder problem! And very hard if  $\mu_a(t)$  can change at any step!

# Piece-wise stationary bandits

## Stationary MAB problems

Arm  $a$  gives rewards sampled from **the same distribution** for any time step

$$\forall t, r_a(t) \stackrel{\text{iid}}{\sim} \nu_a = \mathcal{B}(\mu_a).$$

## Non stationary MAB problems?

(possibly) **different distributions** for any time step !

$$\forall t, r_a(t) \stackrel{\text{iid}}{\sim} \nu_a(t) = \mathcal{B}(\mu_a(t)).$$

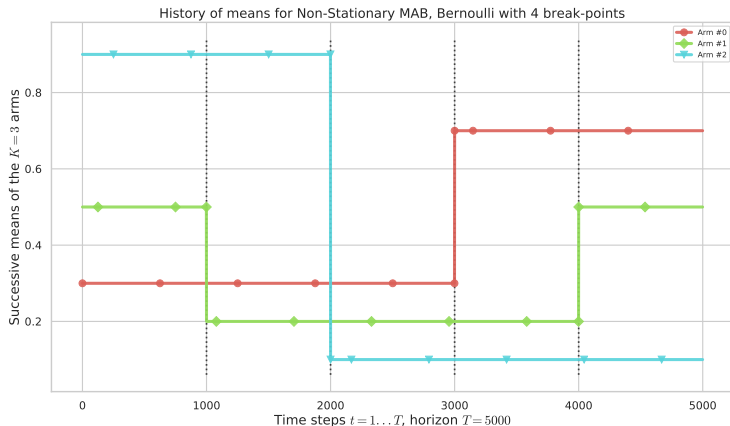
$\Rightarrow$  harder problem! And very hard if  $\mu_a(t)$  can change at any step!

## Piece-wise stationary problems!

$\Leftrightarrow$  the litterature usually focuses on the easier case, when there are at most  $Y_T = o(\sqrt{T})$  intervals, on which the means are all stationary.

# Example of a piece-wise stationary MAB problem

We plot the means  $\mu_1(t)$ ,  $\mu_2(t)$ ,  $\mu_3(t)$  of  $K = 3$  arms. There are  $Y_T = 4$  break-points and 5 sequences between  $t = 1$  and  $t = T = 5000$ :



## Regret for piece-wise stationary bandits

The “oracle” algorithm plays the (unknown) best arm

$k^*(t) = \operatorname{argmax}_k \mu_k(t)$  (which changes between the  $Y_T \geq 1$  stationary sequences)

$$\mathcal{R}(\mathcal{A}, T) = \mathbb{E} \left[ \sum_{t=1}^T r_{k^*(t)}(t) \right] - \sum_{t=1}^T \mathbb{E} [r(t)] = \left( \sum_{t=1}^T \max_k \mu_k(t) \right) - \sum_{t=1}^T \mathbb{E} [r(t)].$$

# Regret for piece-wise stationary bandits

The “oracle” algorithm plays the (unknown) best arm

$k^*(t) = \operatorname{argmax}_k \mu_k(t)$  (which changes between the  $Y_T \geq 1$  stationary sequences)

$$\mathcal{R}(\mathcal{A}, T) = \mathbb{E} \left[ \sum_{t=1}^T r_{k^*(t)}(t) \right] - \sum_{t=1}^T \mathbb{E} [r(t)] = \left( \sum_{t=1}^T \max_k \mu_k(t) \right) - \sum_{t=1}^T \mathbb{E} [r(t)].$$

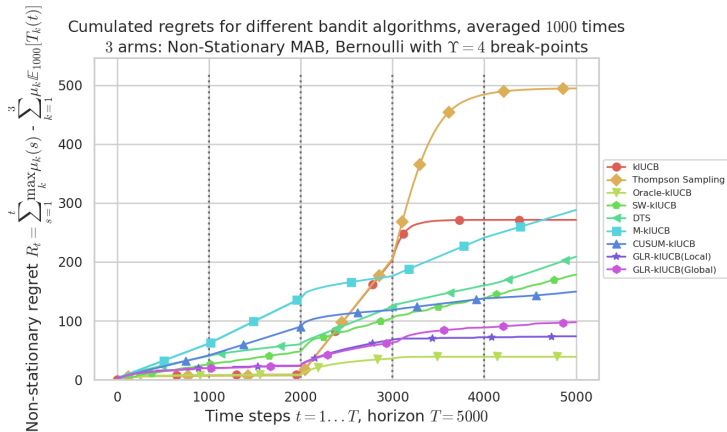
## Typical regimes for piece-wise stationary bandits

- ▶ The lower-bound is  $\mathcal{R}(\mathcal{A}, T) \geq \Omega(\sqrt{KTY_T})$
- ▶ Currently, state-of-the-art algorithms  $\mathcal{A}$  obtain
  - ▶  $\mathcal{R}(\mathcal{A}, T) \leq \mathcal{O}(K\sqrt{TY_T \log(T)})$  if  $T$  and  $Y_T$  are known
  - ▶  $\mathcal{R}(\mathcal{A}, T) \leq \mathcal{O}(KY_T\sqrt{T \log(T)})$  if  $T$  and  $Y_T$  are unknown
- ▶  $\leftrightarrow$  our algorithm **klUCB index + BGLR detector** is state-of-the-art!

[Besson and Kaufmann, 19] arXiv:1902.01575

# Results on a piece-wise stationary MAB problem

**Idea:** combine a good *bandit algorithm* with an *break-point detector*



**kIUCB + BGLR** achieves the best performance (among non-oracle)!

# Other Bandit Models

Many different extensions

Piece-wise stationary bandits

**Multi-player bandits**



# Multi-players bandits: setup

$M$  players playing the same  $K$ -armed bandit ( $2 \leq M \leq K$ )

At round  $t$ :

- ▶ player  $m$  selects  $A_{m,t}$  ; then observes  $X_{A_{m,t},t}$
- ▶ and receives the reward

$$X_{m,t} = \begin{cases} X_{A_{m,t},t} & \text{if no other player chose the same arm} \\ 0 & \text{else (= collision)} \end{cases}$$

**Goal:**

- ▶ maximize centralized rewards  $\sum_{m=1}^M \sum_{t=1}^T X_{m,t}$
- ▶ ... **without communication** between players
- ▶ trade off : exploration / exploitation / **and collisions !**

Cognitive radio: (OSA) sensing, attempt of transmission if no PU,  
possible collisions with other SUs ↪ see the next talk at 4pm !

**Idea:** combine a good *bandit algorithm* with an *orthogonalization strategy* (collision avoidance protocol)

**Example:** UCB1 +  $\rho^{\text{rand}}$ . At round  $t$  each player

- ▶ has a stored rank  $R_{m,t} \in \{1, \dots, M\}$
- ▶ selects the arm that has **the  $R_{m,t}$ -largest UCB**
- ▶ if a collision occurs, draws a new rank  $R_{m,t+1} \sim \mathcal{U}(\{1, \dots, M\})$
- ▶ any index policy may be used in place of UCB1
- ▶ **their proof was wrong...**
- ▶ **Early references:** [Liu and Zhao, 10] [Anandkumar et al., 11]

**Idea:** combine a good *bandit algorithm* with an *orthogonalization strategy* (collision avoidance protocol)

**Example:** our algorithm **klUCB index + MC-TopM rule**

- ▶ more complicated behavior (musical chair game)
- ▶ we obtain a  $\mathcal{R}(\mathcal{A}, T) = \mathcal{O}(M^3 \frac{1}{\Delta_M^2} \log(T))$  regret upper bound
- ▶ lower bound is  $\mathcal{R}(\mathcal{A}, T) = \Omega(M \frac{1}{\Delta_M^2} \log(T))$
- ▶ order optimal, not asymptotically optimal
- ▶ **Recent references:** [Besson and Kaufmann, 18] [Boursier et al, 19]

# Multi-players bandits: algorithms

**Idea:** combine a good *bandit algorithm* with an *orthogonalization strategy* (collision avoidance protocol)

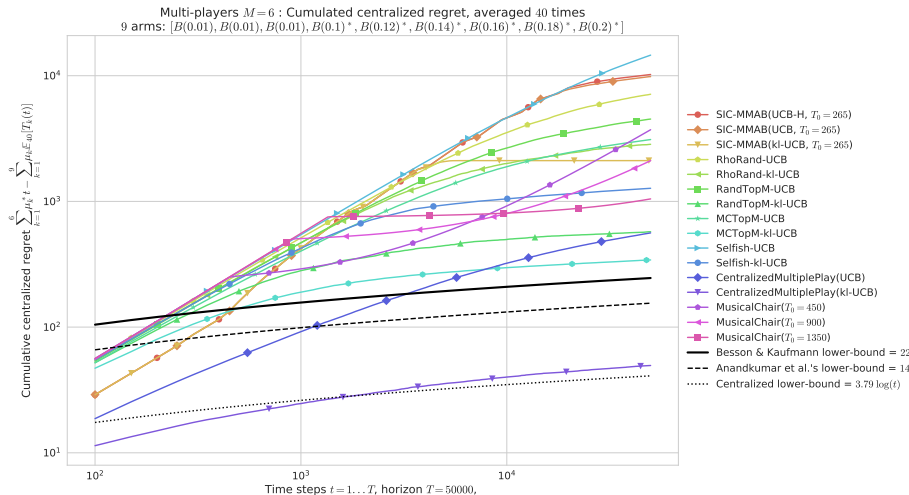
**Example:** our algorithm **klUCB index + MC-TopM rule**

▶ **Recent references:** [Besson and Kaufmann, 18] [Boursier et al, 19]

## Remarks:

- ▶ number of players  $M$  has to be known  
⇒ but it is possible to estimate it on the run
- ▶ does not handle an evolving number of devices (entering/leaving the network)
- ▶ is it a *fair* orthogonalization rule?
- ▶ could players use the collision indicators to communicate? (yes!)

# Results on a multi-player MAB problem



For  $M = 6$  objects, our strategy (MC-TopM) largely outperform SIC-MMAB and  $\rho^{\text{rand}}$ .  
**MCTopM + klUCB** achieves the best performance (among decentralized algorithms) !

# SUMMARY

Now you are aware of:

- ▶ several methods for facing an exploration/exploitation dilemma
- ▶ notably two powerful classes of methods
  - ▶ optimistic “UCB” algorithms
  - ▶ Bayesian approaches, mostly Thompson Sampling

⇒ And you can learn more about [more complex bandit problems](#) and [Reinforcement Learning](#)!

You also saw a bunch of **important tools**:

- ▶ performance lower bounds, guiding the design of algorithms
- ▶ Kullback-Leibler divergence to measure deviations
- ▶ applications of self-normalized concentration inequalities
- ▶ Bayesian tools. . .

And we presented many extensions of the single-player stationary MAB model.



Check out the

## “The Bandit Book”

by Tor Lattimore and Csaba Szepesvári  
Cambridge University Press, 2019.

↪ [tor-lattimore.com/downloads/book/book.pdf](http://tor-lattimore.com/downloads/book/book.pdf)

Reach me (or Émilie Kaufmann) out by email, if you have questions

Lilian.Besson @ Inria.fr

↪ [perso.crans.org/besson/](http://perso.crans.org/besson/)

Emilie.Kaufmann @ Univ-Lille.fr

↪ [chercheurs.lille.inria.fr/ekaufman](http://chercheurs.lille.inria.fr/ekaufman)

Experiment with bandits by yourself!

Interactive demo on this web-page

↪ [perso.crans.org/besson/phd/MAB\\_interactive\\_demo/](https://perso.crans.org/besson/phd/MAB_interactive_demo/)


Use our Python library for simulations of MAB problems **SMPyBandits**

↪ [SMPyBandits.GitHub.io](https://SMPyBandits.GitHub.io) & [GitHub.com/SMPyBandits](https://GitHub.com/SMPyBandits)

- ▶ Install with `$ pip install SMPyBandits`
- ▶ Free and open-source (MIT license)
- ▶ Easy to set up your own bandit experiments, add new algorithms etc.

Welcome to SMPyBandits
+

← → 🏠
🔒 <https://smpybandits.github.io/index.html>
🔍 ⋮



0.9

**CONTENTS:**

- SMPyBandits
- SMPyBandits modules
- How to run the code ?
- List of research publications using Lilian Besson's SMPyBandits project
- Policy aggregation algorithms
- Multi-players simulation environment
- Doubling Trick for Multi-Armed Bandits
- Structure and Sparsity of Stochastic Multi-Armed Bandits
- Non-Stationary Stochastic Multi-Armed Bandits
- Short documentation of the API
- ★ TODO
- Some illustrations for this project
- Jupyter Notebooks 📖 by Naereen @ GitHub
- List of notebooks for SMPyBandits
- A note on execution times, speed and profiling
- UML diagrams
- Logs files

## Welcome to SMPyBandits documentation!

Open-Source Python package for Single- and Multi-Players multi-armed Bandits algorithms.

A research framework for Single and Multi-Players Multi-Arms Bandits (MAB) Algorithms: UCB, KL-UCB, Thompson and many more for single-players, and MCTopM & RandTopM, MusicalChair, ALOHA, MEGA, rhoRand for multi-players simulations. It runs on Python 2 and 3, and is publically released as an open-source software under the MIT License.

**Note**

See more on the [GitHub page](https://github.com/SMPyBandits/SMPyBandits/) for this project: <https://github.com/SMPyBandits/SMPyBandits/>. The project is also hosted on [Inria GForge](https://inria.github.io), and the documentation can be seen online at <https://smpybandits.github.io/> or <http://banditslilian.gforge.inria.fr/> or <https://smpybandits.readthedocs.io/> [webSite](#) [cd](#)

This repository contains the code of my numerical environment, written in Python, in order to perform numerical simulations on single-player and multi-players Multi-Armed Bandits (MAB) algorithms.

🔗 Open Source? Yes!
🔗 Maintained? yes
🗨 Ask me anything
📦 pip! v0.9.5
🔗 Implementation: cpython

🐍 python 2.7 | 3.4 | 3.5 | 3.6
📖 docs: passing bulletpassing

I (Lilian Besson) have started my PhD in October 2016, and this is a part of my on going research since December 2016.

### How to cite this work?

If you use this package for your own work, please consider citing it with this piece of BibTeX:

```

@misc{SMPyBandits,
  title = {{SMPyBandits: an Open-Source Research Framework for S
  author = {Lilian Besson},
  year = {2018},
  url = {https://github.com/SMPyBandits/SMPyBandits/},
  howpublished = {Online at: \url{github.com/SMPyBandits/SMPyBandi
  note = {Code at https://github.com/SMPyBandits/SMPyBandits/,
  }

```

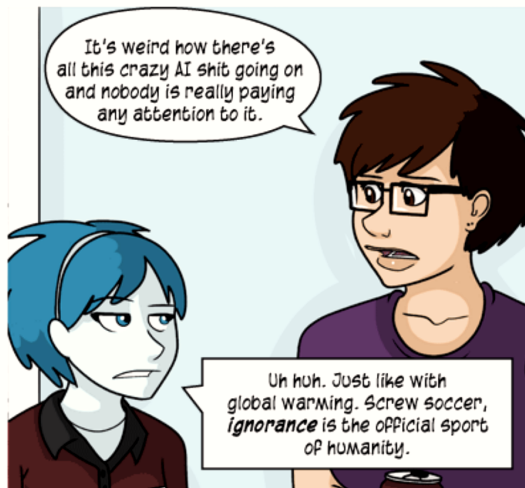
**Thanks for your attention !**

Questions & Discussion ?

# Thanks for your attention !

## Questions & Discussion ?

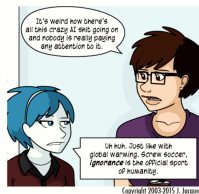
↪ Break and then next talk by Christophe Moy  
*“Decentralized Spectrum Learning for IoT”*



Copyright 2003-2015 J. Jacques

© Jeph Jacques, 2015, [QuestionableContent.net/view.php?comic=3074](http://QuestionableContent.net/view.php?comic=3074)

# Let's talk about actions against the climatic crisis !



## We are *scientists*. . .

Goals: **inform ourselves, think, find, communicate!**

- ▶ **Inform ourselves** of the **causes** and **consequences** of climatic crisis,
- ▶ **Think** of the all the problems, at political, local and individual scales,
- ▶ **Find** simple solutions !
  - ⇒ Aim at sobriety: transports, tourism, clothing, food, computations, fighting smoking, etc.
- ▶ **Communicate** our awareness, and our actions !



# Main references

- ▶ My PhD thesis (Lilian Besson)  
“Multi-players Bandit Algorithms for Internet of Things Networks”  
↪ [perso.crans.org/besson/phd/](http://perso.crans.org/besson/phd/)  
↪ [GitHub.com/Naareen/phd-thesis/](https://github.com/Naareen/phd-thesis/)
- ▶ Our Python library for simulations of MAB problems, **SMPyBandits**  
↪ [SMPyBandits.GitHub.io](https://github.com/SMPyBandits)
- ▶ “The Bandit Book”, by Tor Lattimore and Csaba Szepesvari  
↪ [tor-lattimore.com/downloads/book/book.pdf](http://tor-lattimore.com/downloads/book/book.pdf)
- ▶ “Introduction to Multi-Armed Bandits”, by Alex Slivkins  
↪ [arXiv.org/abs/1904.07272](https://arxiv.org/abs/1904.07272)

- ▶ W.R. Thompson (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*.
- ▶ H. Robbins (1952). Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*.
- ▶ Bradt, R., Johnson, S., and Karlin, S. (1956). On sequential designs for maximizing the sum of  $n$  observations. *Annals of Mathematical Statistics*.
- ▶ R. Bellman (1956). A problem in the sequential design of experiments. *The indian journal of statistics*.
- ▶ Gittins, J. (1979). Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society*.
- ▶ Berry, D. and Fristedt, B. Bandit Problems (1985). Sequential allocation of experiments. *Chapman and Hall*.
- ▶ Lai, T. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*.
- ▶ Lai, T. (1987). Adaptive treatment allocation and the multi-armed bandit problem. *Annals of Statistics*.

## References (2/6)

- ▶ Agrawal, R. (1995). Sample mean based index policies with  $\mathcal{O}(\log n)$  regret for the multi-armed bandit problem. *Advances in Applied Probability*.
- ▶ Katehakis, M. and Robbins, H. (1995). Sequential choice from several populations. *Proceedings of the National Academy of Science*.
- ▶ Burnetas, A. and Katehakis, M. (1996). Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics*.
- ▶ Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*.
- ▶ Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. (2002). The nonstochastic multiarmed bandit problem. *SIAM Journal of Computing*.
- ▶ Burnetas, A. and Katehakis, M. (2003). Asymptotic Bayes Analysis for the finite horizon one armed bandit problem. *Probability in the Engineering and Informational Sciences*.
- ▶ Cesa-Bianchi, N. and Lugosi, G. (2006). Prediction, Learning and Games. *Cambridge University Press*.
- ▶ Audibert, J-Y., Munos, R. and Szepesvari, C. (2009). Exploration-exploitation trade-off using variance estimates in multi-armed bandits. *Theoretical Computer Science*.

- ▶ Audibert, J.-Y. and Bubeck, S. (2010). Regret Bounds and Minimax Policies under Partial Monitoring. *Journal of Machine Learning Research*.
- ▶ Li, L., Chu, W., Langford, J. and Shapire, R. (2010). A Contextual-Bandit Approach to Personalized News Article Recommendation. *WWW*.
- ▶ Honda, J. and Takemura, A. (2010). An Asymptotically Optimal Bandit Algorithm for Bounded Support Models. *COLT*.
- ▶ Bubeck, S. (2010). Jeux de bandits et fondation du clustering. PhD thesis, Université de Lille 1.
- ▶ A. Anandkumar, N. Michael, A. K. Tang, and S. Agrawal (2011). Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*
- ▶ Garivier, A. and Cappé, O. (2011). The KL-UCB algorithm for bounded stochastic bandits and beyond. *COLT*.
- ▶ Maillard, O.-A., Munos, R., and Stoltz, G. (2011). A Finite-Time Analysis of Multi-armed Bandits Problems with Kullback-Leibler Divergences. *COLT*.
- ▶ Chapelle, O. and Li, L. (2011). An empirical evaluation of Thompson Sampling. *NIPS*.

- ▶ E. Kaufmann, O. Cappé, A. Garivier (2012). On Bayesian Upper Confidence Bounds for Bandits Problems. *AISTATS*.
- ▶ Agrawal, S. and Goyal, N. (2012). Analysis of Thompson Sampling for the multi-armed bandit problem. *COLT*.
- ▶ E. Kaufmann, N. Korda, R. Munos (2012), Thompson Sampling : an Asymptotically Optimal Finite-Time Analysis. *Algorithmic Learning Theory*.
- ▶ Bubeck, S. and Cesa-Bianchi, N. (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*.
- ▶ Agrawal, S. and Goyal, N. (2013). Further Optimal Regret Bounds for Thompson Sampling. *AISTATS*.
- ▶ O. Cappé, A. Garivier, O-A. Maillard, R. Munos, and G. Stoltz (2013). Kullback-Leibler upper confidence bounds for optimal sequential allocation. *Annals of Statistics*.
- ▶ Korda, N., Kaufmann, E., and Munos, R. (2013). Thompson Sampling for 1-dimensional Exponential family bandits. *NIPS*.

- ▶ Honda, J. and Takemura, A. (2014). Optimality of Thompson Sampling for Gaussian Bandits depends on priors. *AISTATS*.
- ▶ Baransi, Maillard, Mannor (2014). Sub-sampling for multi-armed bandits. *ECML*.
- ▶ Honda, J. and Takemura, A. (2015). Non-asymptotic analysis of a new bandit algorithm for semi-bounded rewards. *JMLR*.
- ▶ Kaufmann, E., Cappé O. and Garivier, A. (2016). On the complexity of best arm identification in multi-armed bandit problems. *JMLR*
- ▶ Lattimore, T. (2016). Regret Analysis of the Finite-Horizon Gittins Index Strategy for Multi-Armed Bandits. *COLT*.
- ▶ Garivier, A., Kaufmann, E. and Lattimore, T. (2016). On Explore-Then-Commit strategies. *NIPS*.
- ▶ E.Kaufmann (2017), On Bayesian index policies for sequential resource allocation. *Annals of Statistics*.
- ▶ Agrawal, S. and Goyal, N. (2017). Near-Optimal Regret Bounds for Thompson Sampling. *Journal of ACM*.

- ▶ Maillard, O-A (2017). Boundary Crossing for General Exponential Families. *Algorithmic Learning Theory*.
- ▶ Besson, L., Kaufmann E. (2018). Multi-Player Bandits Revisited. *Algorithmic Learning Theory*.
- ▶ Cowan, W., Honda, J. and Katehakis, M.N. (2018). Normal Bandits of Unknown Means and Variances. *JMLR*.
- ▶ Garivier, A. and Ménard, P. and Stoltz, G. (2018). Explore first, exploit next: the true shape of regret in bandit problems, *Mathematics of Operations Research*
- ▶ Garivier, A. and Hadiji, H. and Ménard, P. and Stoltz, G. (2018). KL-UCB-switch: optimal regret bounds for stochastic bandits from both a distribution-dependent and a distribution-free viewpoints. *arXiv: 1805.05071*.
- ▶ Besson, L., Kaufmann E. (2019). The Generalized Likelihood Ratio Test meets klUCB: an Improved Algorithm for Piece-Wise Non-Stationary Bandits. *Algorithmic Learning Theory*. *arXiv: 1902.01575*.