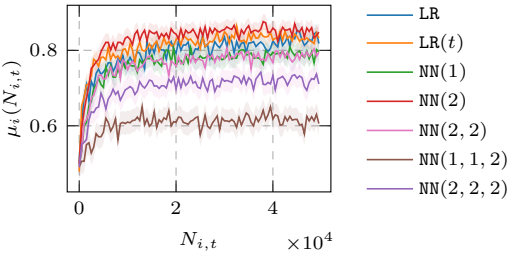
Stochastic Rising Bandits

Alberto Maria Metelli Francesco Trovò Matteo Pirola Marcello Restelli

20 July 2022
The Thirty-ninth International Conference on Machine Learning (ICML 2022)

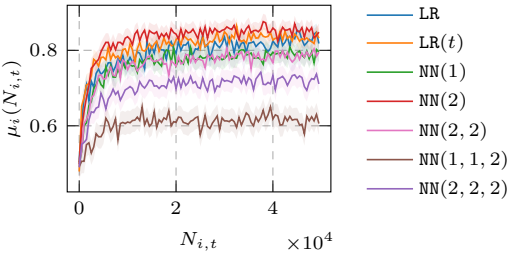
Motivation: Online Model Selection

- **Problem**: choose among a set of **learning algorithms**
- Performance of each algorithm **increases** on average
- At each round, you can assign a unit of resources (e.g., computational power, samples) to a single algorithm
- **Goal**: find the **"best"** learning algorithm



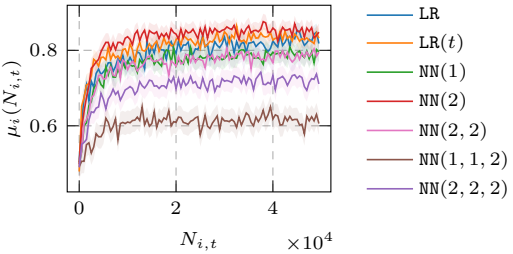
Motivation: Online Model Selection

- **Problem**: choose among a set of **learning algorithms**
- Performance of each algorithm **increases** on average
- At each round, you can assign a unit of resources (e.g., computational power, samples) to a single algorithm
- **Goal**: find the **"best"** learning algorithm



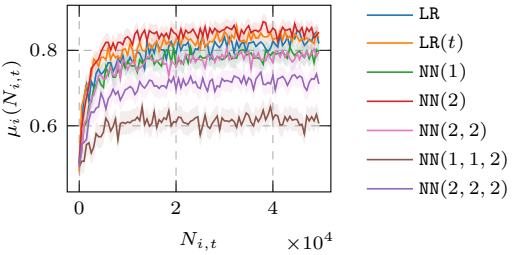
Motivation: Online Model Selection

- **Problem**: choose among a set of **learning algorithms**
- Performance of each algorithm **increases** on average
- At each round, you can assign a unit of resources (e.g., computational power, samples) to a single algorithm
- **Goal**: find the **"best"** learning algorithm

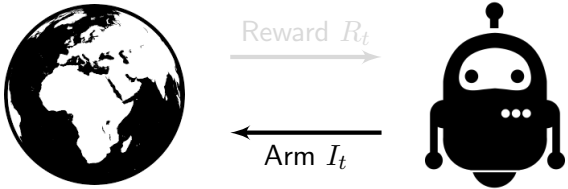


Motivation: Online Model Selection

- **Problem**: choose among a set of **learning algorithms**
- Performance of each algorithm **increases** on average
- At each round, you can assign a unit of resources (e.g., computational power, samples) to a single algorithm
- **Goal**: find the **"best"** learning algorithm



Multi-Armed Bandits (Lattimore and Szepesvári, 2020)



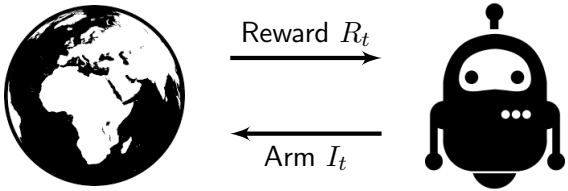
- Agent plays an **arm**
$$I_t \in [K] := \{1, \dots, K\}$$
- Environment generates a **reward**
$$R_t \sim \nu_{I_t}(t, N_{I_t,t})$$
with **expected reward** $\mu_i(t, N_{I_t,t})$

If the (expected) reward depends on the (Seznec et al., 2020)

- current **round** $t \in [T] \implies$ **restless** arm $\mu_i(t)$
- **number of pulls** to the arm $N_{i,t} = \sum_{l=1}^{t} \mathbb{1}\{I_l = i\} \implies$ **rested** arm $\mu_i(N_{i,t})$

We will focus on the **rested** *setting*

Multi-Armed Bandits (Lattimore and Szepesvári, 2020)



- Agent plays an **arm**
$$I_t \in [K] \coloneqq \{1, \dots, K\}$$
- Environment generates a **reward**
$$R_t \sim \nu_{I_t}(t, N_{I_t,t})$$

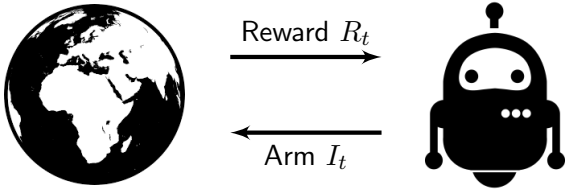
with **expected reward** $\mu_i(t, N_{I_t,t})$

If the (expected) reward depends on the (Seznec et al., 2020)

- current **round** $t \in [T] \implies$ **restless** arm $\mu_i(t)$
- **number of pulls** to the arm $N_{i,t} = \sum_{l=1}^{t} \mathbb{1}\{I_l = i\} \implies$ **rested** arm $\mu_i(N_{i,t})$

We will focus on the **rested** *setting*

Multi-Armed Bandits (Lattimore and Szepesvári, 2020)



- Agent plays an **arm**
$$I_t \in [K] \coloneqq \{1, \dots, K\}$$
- Environment generates a **reward**
$$R_t \sim \nu_{I_t}(t, N_{I_t,t})$$

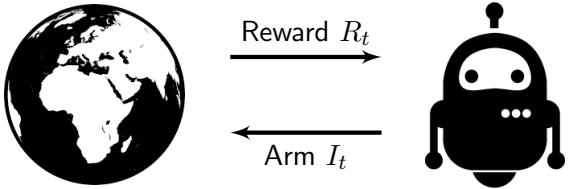
with **expected reward** $\mu_i(t, N_{I_t,t})$

If the (expected) reward depends on the (Seznec et al., 2020)

- current **round** $t \in [T] \implies$ **restless** arm $\mu_i(t)$
- **number of pulls** to the arm $N_{i,t} = \sum_{l=1}^{t} \mathbb{1}\{I_l = i\} \implies$ **rested** arm $\mu_i(N_{i,t})$

We will focus on the **rested** *setting*

Multi-Armed Bandits (Lattimore and Szepesvári, 2020)



- Agent plays an **arm**
$$I_t \in [K] := \{1, \dots, K\}$$
- Environment generates a **reward**
$$R_t \sim \nu_{I_t}(t, N_{I_t,t})$$

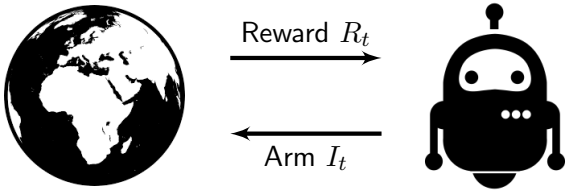
with **expected reward** $\mu_i(t, N_{I_t,t})$

If the (expected) reward depends on the (Seznec et al., 2020)

- current **round** $t \in [T] \implies$ **restless** arm $\mu_i(t)$
- **number of pulls** to the arm $N_{i,t} = \sum_{l=1}^{t} \mathbb{1}\{I_l = i\} \implies$ **rested** arm $\mu_i(N_{i,t})$

We will focus on the **rested** *setting*

Multi-Armed Bandits (Lattimore and Szepesvári, 2020)



- Agent plays an **arm**
$$I_t \in [K] := \{1, \dots, K\}$$
- Environment generates a **reward**
$$R_t \sim \nu_{I_t}(t, N_{I_t,t})$$

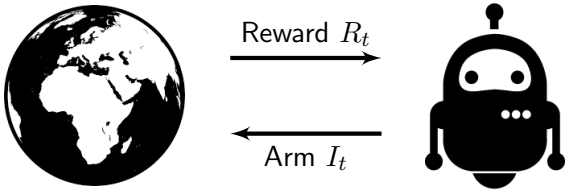
with **expected reward** $\mu_i(t, N_{I_t,t})$

If the (expected) reward depends on the (Seznec et al., 2020)

- current **round** $t \in [T] \implies$ **restless** arm $\mu_i(t)$
- **number of pulls** to the arm $N_{i,t} = \sum_{l=1}^{t} \mathbb{1}\{I_l = i\} \implies$ **rested** arm $\mu_i(N_{i,t})$

We will focus on the **rested** *setting*

Multi-Armed Bandits (Lattimore and Szepesvári, 2020)



- Agent plays an **arm**
$$I_t \in [K] := \{1, \dots, K\}$$
- Environment generates a **reward**
$$R_t \sim \nu_{I_t}(t, N_{I_t,t})$$

with **expected reward** $\mu_i(t, N_{I_t,t})$

If the (expected) reward depends on the (Seznec et al., 2020)

- current **round** $t \in [T] \implies$ **restless** arm $\mu_i(t)$
- **number of pulls** to the arm $N_{i,t} = \sum_{l=1}^{t} \mathbb{1}\{I_l = i\} \implies$ **rested** arm $\mu_i(N_{i,t})$

We will focus on the **rested** *setting*

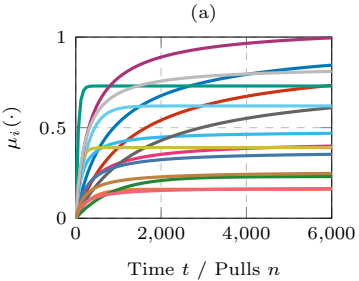
Rising Bandits (Heidari et al., 2016)

- **Non-Decreasing** Assumption $\implies$ *"increase of total return"*

$$\gamma_i(n) := \mu_i(n+1) - \mu_i(n) \geqslant 0$$

- **Concavity** Assumption $\implies$ *"decrease of marginal returns"*

$$\gamma_i(n+1) - \gamma_i(n) \leqslant 0$$



Goal: find a policy π^* minimizing the **policy regret** (Dekel et al., 2012):

$$R(\pi, T) := \max_{\pi} J(\pi, T) - J(\pi, T) \qquad J(\pi, T) := \mathop{\mathbb{E}}_{\pi}\Big[\sum_{t\in[T]} \mu_{I_t}(N_{I_t,t})\Big]$$

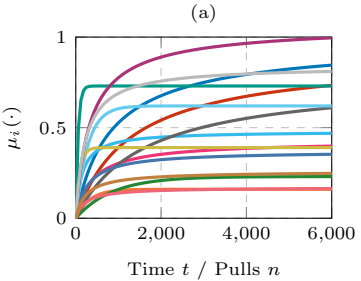
Rising Bandits (Heidari et al., 2016)

- **Non-Decreasing** Assumption $\implies$ *"increase of total return"*

$$\gamma_i(n) := \mu_i(n+1) - \mu_i(n) \geqslant 0$$

- **Concavity** Assumption $\implies$ *"decrease of marginal returns"*

$$\gamma_i(n+1) - \gamma_i(n) \leqslant 0$$



Goal: find a policy π^* minimizing the **policy regret** (Dekel et al., 2012):

$$R(\pi, T) := \max_{\pi} J(\pi, T) - J(\pi, T) \qquad J(\pi, T) := \mathop{\mathbb{E}}_{\pi}\Big[\sum_{t\in[T]} \mu_{I_t}(N_{I_t,t})\Big]$$

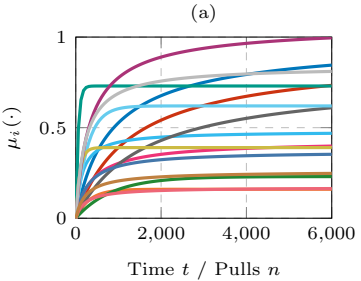
Rising Bandits (Heidari et al., 2016)

- **Non-Decreasing** Assumption $\implies$ *"increase of total return"*

$$\gamma_i(n) := \mu_i(n+1) - \mu_i(n) \geqslant 0$$

- **Concavity** Assumption $\implies$ *"decrease of marginal returns"*

$$\gamma_i(n+1) - \gamma_i(n) \leqslant 0$$



Goal: find a policy π^* minimizing the **policy regret** (Dekel et al., 2012):

$$R(\pi, T) := \max_{\pi} J(\pi, T) - J(\pi, T) \qquad J(\pi, T) := \mathop{\mathbb{E}}_{\pi}\Big[\sum_{t\in[T]} \mu_{I_t}(N_{I_t,t})\Big]$$

Estimator Construction: Deterministic Case

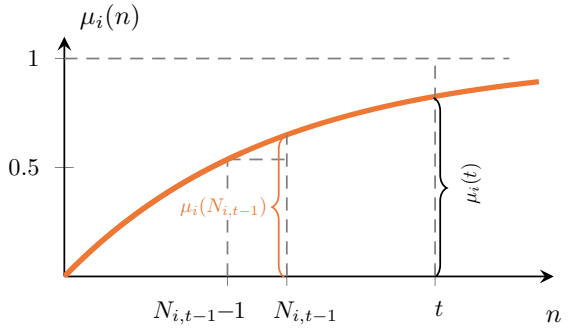
- **Idea**: exploit the **concavity** assumption to get an **optimistic** estimate of $\mu_i(t)$:

$$\mu_i(t) = \underbrace{\mu_i(N_{i,t-1})}_{\text{(most recent payoff)}} + \underbrace{\sum_{n=N_{i,t-1}}^{t-1} \gamma_i(n)}_{\text{(sum of future increments)}} \leqslant \underbrace{\mu_i(N_{i,t-1})}_{\text{(most recent payoff)}} + (t - N_{i,t-1}) \underbrace{\gamma_i(N_{i,t-1} - 1)}_{\text{(most recent increment)}}$$

Estimator Construction: Deterministic Case

- **Idea**: exploit the **concavity** assumption to get an **optimistic** estimate of $\mu_i(t)$:

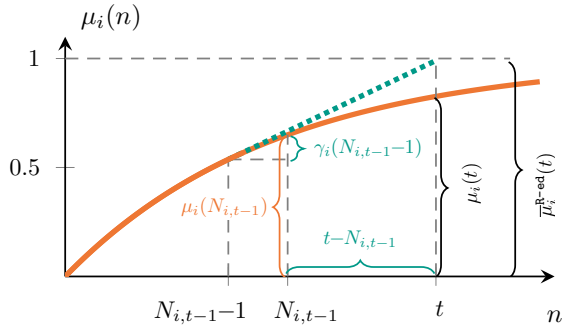
$$\mu_i(t) = \underbrace{\mu_i(N_{i,t-1})}_{\text{(most recent payoff)}} + \underbrace{\sum_{n=N_{i,t-1}}^{t-1} \gamma_i(n)}_{\text{(sum of future increments)}} \leqslant \underbrace{\mu_i(N_{i,t-1})}_{\text{(most recent payoff)}} + (t - N_{i,t-1}) \underbrace{\gamma_i(N_{i,t-1} - 1)}_{\text{(most recent increment)}}$$



Estimator Construction: Deterministic Case

- **Idea**: exploit the **concavity** assumption to get an **optimistic** estimate of $\mu_i(t)$:

$$\mu_i(t) = \underbrace{\mu_i(N_{i,t-1})}_{\text{(most recent payoff)}} + \underbrace{\sum_{n=N_{i,t-1}}^{t-1} \gamma_i(n)}_{\text{(sum of future increments)}} \leqslant \underbrace{\mu_i(N_{i,t-1})}_{\text{(most recent payoff)}} + (t - N_{i,t-1}) \underbrace{\gamma_i(N_{i,t-1} - 1)}_{\text{(most recent increment)}}$$



Estimator Construction: Stochastic Case

- **Problem**: we observe only **noisy** versions of the expected reward $\mu_i(n)$
- **Idea**: average over a **window** of h pulls ($h \leqslant \lfloor N_{i,t-1}/2 \rfloor$)
  - **Bias-variance** trade-off choice

$$\widehat{\mu}_i^h(t) := \frac{1}{h} \sum_{t=N_{i,t-1}-h+1}^{N_{i,t-1}} \Bigg(\underbrace{R_{i,t}}_{\text{(estimated payoff)}} + (t-1) \underbrace{\frac{R_{i,t} - R_{i,t-h}}{h}}_{\text{(estimated increment)}} \Bigg)$$

- Add a **bonus** term to account for the rewards concentration $\beta_i^h(t, \delta_t)$
- Play the **optimistic** arm $I_t \in \arg\max_{i \in [K]} \widehat{\mu}_i^h(t) + \beta_i^h(t, \delta_t) \implies$ `R-ed-UCB`

Estimator Construction: Stochastic Case

- **Problem**: we observe only **noisy** versions of the expected reward $\mu_i(n)$
- **Idea**: average over a **window** of h pulls ($h \leqslant \lfloor N_{i,t-1}/2 \rfloor$)
  - **Bias-variance** trade-off choice

$$\widehat{\mu}_i^h(t) := \frac{1}{h} \sum_{l=N_{i,t-1}-h+1}^{N_{i,t-1}} \left(\underbrace{R_{t_{i,l}}}_{\text{(estimated payoff)}} + (t-l) \underbrace{\frac{R_{t_{i,l}} - R_{t_{i,l-h}}}{h}}_{\text{(estimated increment)}} \right)$$

- Add a **bonus** term to account for the rewards concentration $\beta_i^h(t, \delta_t)$
- Play the **optimistic** arm $I_t \in \arg\max_{i \in [K]} \widehat{\mu}_i^h(t) + \beta_i^h(t, \delta_t) \implies$ `R-ed-UCB`

Estimator Construction: Stochastic Case

- **Problem**: we observe only **noisy** versions of the expected reward $\mu_i(n)$
- **Idea**: average over a **window** of h pulls ($h \leqslant \lfloor N_{i,t-1}/2 \rfloor$)
  - **Bias-variance** trade-off choice

$$\widehat{\mu}_i^h(t) := \frac{1}{h} \sum_{l=N_{i,t-1}-h+1}^{N_{i,t-1}} \left(\underbrace{R_{t_{i,l}}}_{\text{(estimated payoff)}} + (t-l) \underbrace{\frac{R_{t_{i,l}} - R_{t_{i,l-h}}}{h}}_{\text{(estimated increment)}} \right)$$

- Add a **bonus** term to account for the rewards concentration $\beta_i^h(t, \delta_t)$
- Play the **optimistic** arm $I_t \in \arg\max_{i \in [K]} \widehat{\mu}_i^h(t) + \beta_i^h(t, \delta_t) \implies$ `R-ed-UCB`

Estimator Construction: Stochastic Case

- **Problem**: we observe only **noisy** versions of the expected reward $\mu_i(n)$
- **Idea**: average over a **window** of h pulls ($h \leqslant \lfloor N_{i,t-1}/2 \rfloor$)
  - **Bias-variance** trade-off choice

$$\widehat{\mu}_i^h(t) := \frac{1}{h} \sum_{l=N_{i,t-1}-h+1}^{N_{i,t-1}} \left(\underbrace{R_{t_{i,l}}}_{\text{(estimated payoff)}} + (t-l) \underbrace{\frac{R_{t_{i,l}} - R_{t_{i,l-h}}}{h}}_{\text{(estimated increment)}} \right)$$

- Add a **bonus** term to account for the rewards concentration $\beta_i^h(t, \delta_t)$
- Play the **optimistic** arm $I_t \in \arg\max_{i \in [K]} \widehat{\mu}_i^h(t) + \beta_i^h(t, \delta_t) \implies$ `R-ed-UCB`

Estimator Construction: Stochastic Case

- **Problem**: we observe only **noisy** versions of the expected reward $\mu_i(n)$
- **Idea**: average over a **window** of h pulls ($h \leqslant \lfloor N_{i,t-1}/2 \rfloor$)
  - **Bias-variance** trade-off choice

$$\widehat{\mu}_i^h(t) := \frac{1}{h} \sum_{l=N_{i,t-1}-h+1}^{N_{i,t-1}} \left(\underbrace{R_{t_{i,l}}}_{\text{(estimated payoff)}} + (t-l) \underbrace{\frac{R_{t_{i,l}} - R_{t_{i,l-h}}}{h}}_{\text{(estimated increment)}} \right)$$

- Add a **bonus** term to account for the rewards concentration $\beta_i^h(t, \delta_t)$
- Play the **optimistic** arm $I_t \in \arg\max_{i \in [K]} \widehat{\mu}_i^h(t) + \beta_i^h(t, \delta_t) \implies$ `R-ed-UCB`

Estimator Construction: Stochastic Case

- **Problem**: we observe only **noisy** versions of the expected reward $\mu_i(n)$
- **Idea**: average over a **window** of h pulls ($h \leqslant \lfloor N_{i,t-1}/2 \rfloor$)
  - **Bias-variance** trade-off choice

$$\widehat{\mu}_i^h(t) := \frac{1}{h} \sum_{l=N_{i,t-1}-h+1}^{N_{i,t-1}} \left(\underbrace{R_{t_{i,l}}}_{\text{(estimated payoff)}} + (t-l) \underbrace{\frac{R_{t_{i,l}} - R_{t_{i,l-h}}}{h}}_{\text{(estimated increment)}} \right)$$

- Add a **bonus** term to account for the rewards concentration $\beta_i^h(t, \delta_t)$
- Play the **optimistic** arm $I_t \in \arg\max_{i \in [K]} \widehat{\mu}_i^h(t) + \beta_i^h(t, \delta_t) \implies$ `R-ed-UCB`

Theorem

Under the assumption that:

- *the window is* **linear** *in the number of pulls* $h = \lfloor \epsilon N_{i,t-1} \rfloor$ *and* $\epsilon \in (0, 1/2)$*;*
- *the confidence term is set to* $\delta_t = t^{-\alpha}$ *and* $\alpha > 2$*;*

for every $q \in [0, 1]$*, the expected regret of* `R-ed-UCB` *is bounded by:*

$$R(\texttt{R-ed-UCB}, T) \leqslant \mathcal{O}\left(\underbrace{\frac{K}{\epsilon}(\sigma T)^{\frac{2}{3}} (\alpha \log T)^{\frac{1}{3}}}_{\text{(problem independent)}} + \underbrace{\frac{KT^q}{1-2\epsilon} \Upsilon\left(\left\lfloor (1-2\epsilon)\frac{T}{K} \right\rfloor, q \right)}_{\text{(problem dependent)}} \right)$$

- $\Upsilon(M, q)$ accounts for the "increasing profile" of the expected reward

$$\Upsilon(M, q) := \max_{i \in [K]} \sum_{l=1}^{M-1} \gamma_i(l)^q$$

Theorem

Under the assumption that:

- *the window is* **linear** *in the number of pulls* $h = \lfloor \epsilon N_{i,t-1} \rfloor$ *and* $\epsilon \in (0, 1/2)$;
- *the confidence term is set to* $\delta_t = t^{-\alpha}$ *and* $\alpha > 2$;

for every $q \in [0, 1]$, *the expected regret of* `R-ed-UCB` *is bounded by:*

$$R(\texttt{R-ed-UCB}, T) \leqslant \mathcal{O}\left(\underbrace{\frac{K}{\epsilon}(\sigma T)^{\frac{2}{3}} (\alpha \log T)^{\frac{1}{3}}}_{\text{(problem independent)}} + \underbrace{\frac{KT^q}{1-2\epsilon} \Upsilon\left(\left\lfloor (1-2\epsilon)\frac{T}{K} \right\rfloor, q \right)}_{\text{(problem dependent)}} \right)$$

- $\Upsilon(M, q)$ accounts for the "increasing profile" of the expected reward

$$\Upsilon(M, q) := \max_{i \in [K]} \sum_{l=1}^{M-1} \gamma_i(l)^q$$

Theorem

Under the assumption that:

- *the window is* **linear** *in the number of pulls* $h = \lfloor \epsilon N_{i,t-1} \rfloor$ *and* $\epsilon \in (0, 1/2)$;
- *the confidence term is set to* $\delta_t = t^{-\alpha}$ *and* $\alpha > 2$;

for every $q \in [0, 1]$, *the expected regret of* `R-ed-UCB` *is bounded by:*

$$R(\texttt{R-ed-UCB}, T) \leqslant \mathcal{O}\left(\underbrace{\frac{K}{\epsilon}(\sigma T)^{\frac{2}{3}} (\alpha \log T)^{\frac{1}{3}}}_{\text{(problem independent)}} + \underbrace{\frac{KT^q}{1-2\epsilon} \Upsilon\left(\left\lceil (1-2\epsilon)\frac{T}{K} \right\rceil, q \right)}_{\text{(problem dependent)}} \right)$$

- $\Upsilon(M, q)$ accounts for the "increasing profile" of the expected reward

$$\Upsilon(M, q) := \max_{i \in [K]} \sum_{l=1}^{M-1} \gamma_i(l)^q$$

Theorem

Under the assumption that:

- *the window is* **linear** *in the number of pulls* $h = \lfloor \epsilon N_{i,t-1} \rfloor$ *and* $\epsilon \in (0, 1/2)$;
- *the confidence term is set to* $\delta_t = t^{-\alpha}$ *and* $\alpha > 2$;

for every $q \in [0, 1]$, *the expected regret of* `R-ed-UCB` *is bounded by:*

$$R(\texttt{R-ed-UCB}, T) \leqslant \mathcal{O}\left(\underbrace{\frac{K}{\epsilon}(\sigma T)^{\frac{2}{3}} (\alpha \log T)^{\frac{1}{3}}}_{\text{(problem independent)}} + \underbrace{\frac{KT^q}{1-2\epsilon} \Upsilon\left(\left\lceil (1-2\epsilon)\frac{T}{K} \right\rceil, q \right)}_{\text{(problem dependent)}} \right)$$

- $\Upsilon(M, q)$ accounts for the "increasing profile" of the expected reward

$$\Upsilon(M, q) := \max_{i \in [K]} \sum_{l=1}^{M-1} \gamma_i(l)^q$$

Conclusions

- What we have done and is not in the presentation (**come to our poster!**)
  - **Restless** setting
  - Experimental results
- What to do next?
  - Regret lower bounds
  - Applications to **federated** learning

Conclusions

- What we have done and is not in the presentation (**come to our poster!**)
  - **Restless** setting
  - Experimental results
- What to do next?
  - Regret lower bounds
  - Applications to **federated** learning

Conclusions

- What we have done and is not in the presentation (**come to our poster!**)
  - **Restless** setting
  - Experimental results
- What to do next?
  - Regret lower bounds
  - Applications to federated learning

Conclusions

- What we have done and is not in the presentation (**come to our poster!**)
  - **Restless** setting
  - Experimental results
- What to do next?
  - Regret **lower bounds**
  - Applications to **federated** learning

Conclusions

- What we have done and is not in the presentation (**come to our poster!**)
  - **Restless** setting
  - Experimental results
- What to do next?
  - Regret **lower bounds**
  - Applications to **federated** learning

Conclusions

- What we have done and is not in the presentation (**come to our poster!**)
  - **Restless** setting
  - Experimental results
- What to do next?
  - Regret **lower bounds**
  - Applications to **federated** learning

Thank You for Your Attention!

Contact: albertomaria.metelli@polimi.it

Code:
github.com/albertometelli/stochastic-rising-bandits

Web page: icml.cc/virtual/2022/spotlight/16244

Poster presentation

Wed 20 Jul 6:30 pm - 8:30 pm @ Hall E

References

O. Dekel, A. Tewari, and R. Arora. Online bandit learning against an adaptive adversary: from regret to policy regret. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

H. Heidari, M. J. Kearns, and A. Roth. Tight policy regret bounds for improving and decaying bandits. In S. Kambhampati, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1562–1570, 2016.

T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

J. Seznec, P. Ménard, A. Lazaric, and M. Valko. A single algorithm for both restless and rested rotting bandits. In *Proceedings of the international conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108, pages 3784–3794, 2020.