Efficient stochastic combinatorial bandits

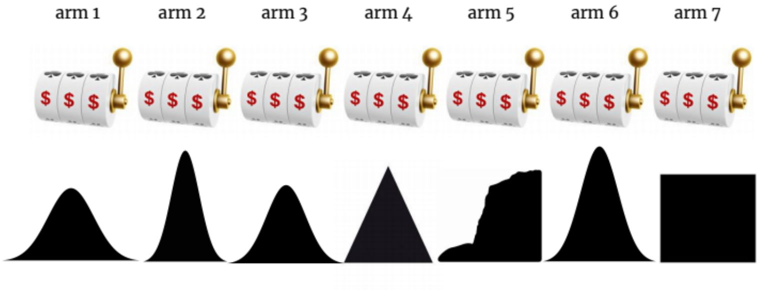
Pierre Perrault

Michal Valko (INRIA Lille) — Vianney Perchet (CMLA, ENS Cachan)

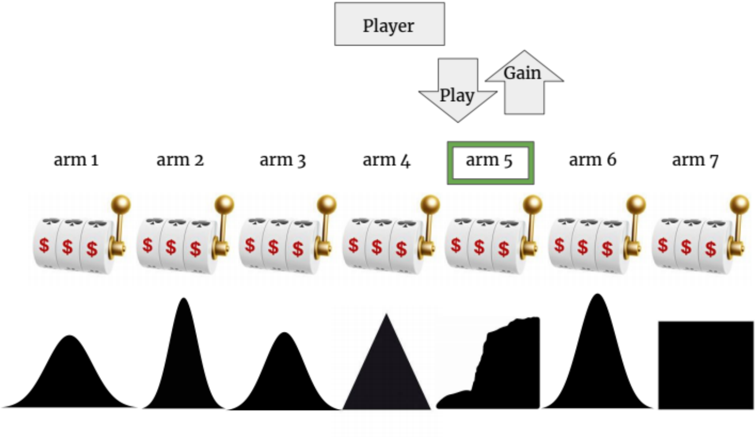
May 21, 2019

Outline

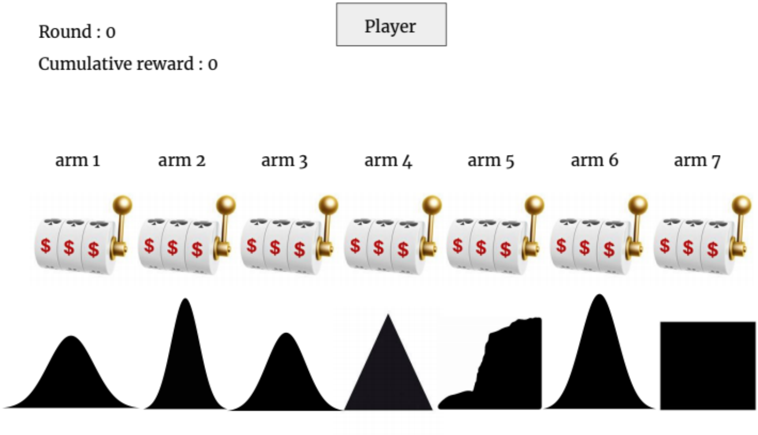
Multi-armed bandit



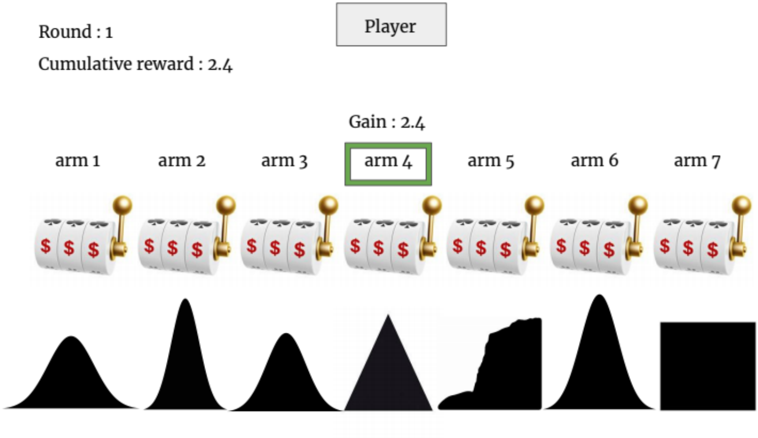
Multi-armed bandit



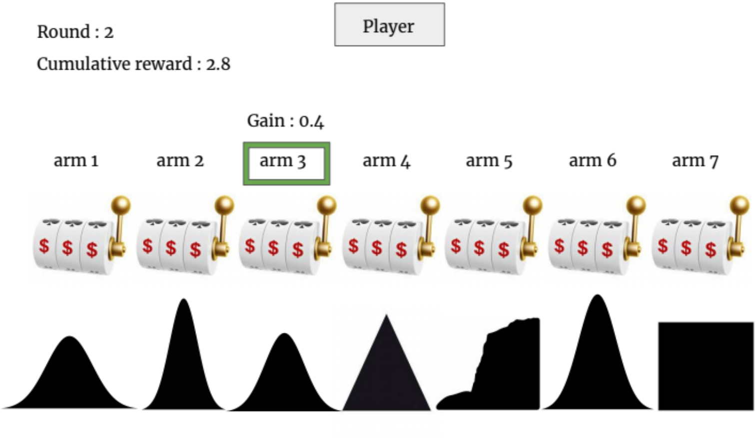
Multi-armed bandit



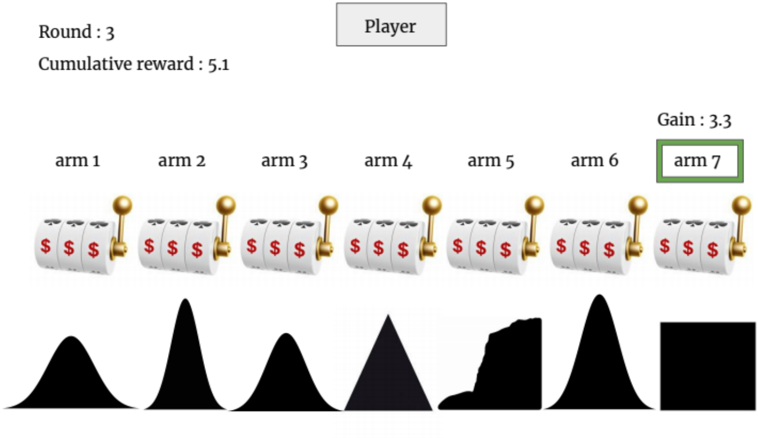
Multi-armed bandit



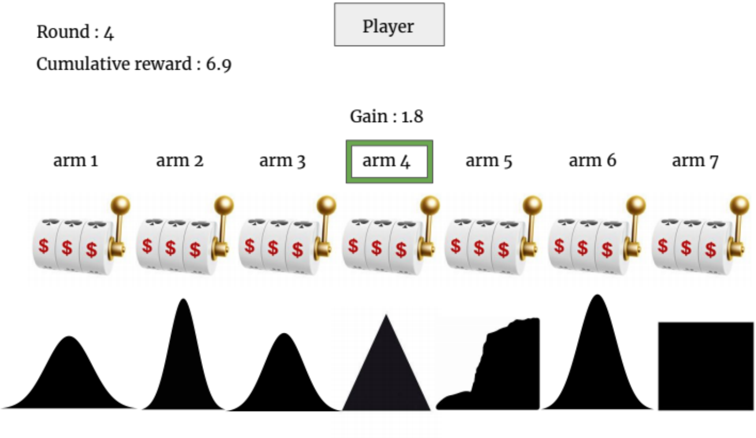
Multi-armed bandit



Multi-armed bandit



Multi-armed bandit



Multi-armed bandit

Rules with n arms, T rounds:
For each round $t \in [T]$:

- Play an arm $i_t \in [n]$
- Receive a reward $X_{i_t,t} \sim$ distribution of arm i_t, **independently from previous rounds rewards.**

Goal : maximize the **expected cumulative reward** :

$$\mathbb{E} \sum_{t \in [T]} X_{i_t,t}$$

or equivalently
minimize the **regret** :

$$R_T \triangleq T\mu^* - \sum_{t \in [T]} \mathbb{E} X_{i_t,t}.$$

with $\mu^* \triangleq \max_i \mu_i, \quad \mu_i \triangleq \mathbb{E} X_i$.

Upper confidence bound (UCB)

At each round t, play $i_t = \text{argmax}_i\ \bar{\mu}_{i,t} + \sqrt{\frac{\log(t)}{T_i}}$.

Where $\mu_{i,t} \triangleq \frac{1}{T_i} \sum_{u \leq t,\ \ i_u = i} X_{i,t}$ and $T_i \triangleq \sum_{u \leq t,\ \ i_u = i} 1.$

Theorem (Auer et al 2002)

$R_T(\textit{UCB}) = O(\frac{\log(T)n}{\Delta})$ *and* $R_T(A) = \Omega(\frac{\log(T)n}{\Delta})\ \forall A$,

with $\Delta \triangleq \min_{i, \mu^* - \mu_i > 0} \mu^* - \mu_i$

Setting

$\mathcal{A} \subset \mathcal{P}([n])$.
At each round t, choose a set $A_t \in \mathcal{A}$.

- Observation : $\{X_{i,t},\ i \in A_t\}$
- Reward : $\sum_{i \in A_t} X_{i,t}$

Regret:

$$R_T = T \sum_{i \in A^*} \mu_i - \sum_{t \in [T]} \mathbb{E} \sum_{i \in A_t} X_{i,t}.$$

Applications

- Online advertising
- Viral marketing — Influence maximization — Fake news propagation
- Recommender system
- Shortest path, rooting in a network

Algorithm CUCB

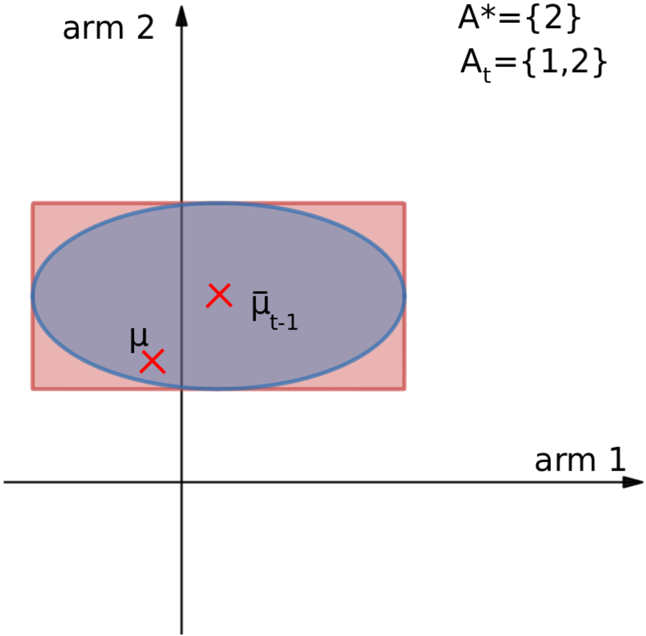
$$A_{t+1} = \text{argmax}_{A \in \mathcal{A}} \sum_{i \in A} \bar{\mu}_{i,t} + \sqrt{\frac{\log(t)}{T_i}}.$$

Theorem (Kveton et al 2015)

$R_T(\textit{CUCB}) = O(\frac{\log(T)mn}{\Delta})$ *and* $R_T(A) = \Omega(\frac{\log(T)mn}{\Delta})\ \forall A$.

- Algorithm is efficient (linear programming at each time step).

What happens if we assume that X_i are mutually independent ?


arm 2
A*={2}
A_t={1,2}
μ̄_{t-1}
μ
arm 1

For X_i mutually independent, the lower bound $R_T(A) = \Omega(\frac{\log(T)mn}{\Delta})\ \forall A$ is no more valid, and we have $R_T(A) = \Omega(\frac{\log(T)n}{\Delta})\ \forall A$

Theorem (Combes et al 2015)

ESCB: $A_{t+1} = \text{argmax}_{A\in\mathcal{A}} A^\top \bar{\mu}_t + \sqrt{\sum_{i\in A} \frac{\log(t)}{T_i}}, \quad R_T = O(\frac{\log(T)n}{\Delta})$.

vs

CUCB: $A_{t+1} = \text{argmax}_{A\in\mathcal{A}} A^\top \bar{\mu}_t + \sum_{i\in A} \sqrt{\frac{\log(t)}{T_i}}, \quad R_T = O(\frac{\log(T)nm}{\Delta})$.

☹ However, ESCB is not efficient...

Proposition

ESCB can be "approximated" efficiently when $\mathcal{A}$ is a matroid.

Matroid

$\mathcal{A}$ is a matroid if:

- $\emptyset \in \mathcal{A}$
- $\mathcal{A}$ is closed under subset
- If $|A| < |B|$ with $A, B \in \mathcal{A}$, then there is $x \in B \backslash A$ s.t. $A \cup \{x\} \in \mathcal{A}$.

Approximation algorithm

$G^* = \max_{A \in \mathcal{A}} G(A)$
$G(A_{approx}) \geq \alpha G^*$, with some $\alpha \in [0, 1]$

Example

Greedy algorithm is a $(1 - 1/e)$-approximation algorithm on matroid when F is submodular, i.e., if $G(A \cap B) + G(A \cup B) \leq G(A) + G(B)$.

Remark

- *Greedy is efficient*
- $G(A) = L(A) + F(A) = A^\top \bar{\mu}_t + \sqrt{\sum_{i \in A} \frac{\log(t)}{T_i}}$ *is submodular.*

☹ However, this approximation leads to linear regret upper bound, since ultimately, $F \to 0$, and the algorithm will not maximize L exactly.

Refined approximation

Theorem

$L(A_{greedy}) + 2F(A_{greedy}) \geq G^*$

This provide the same regret bound, up to a factor 2.

Extensions:

- Local Search approximation algorithm
- Budgeted setting, i.e., actions are costly.

MERCI !