Announcements

- All course materials will be on Course Website, so no need to worry about Canvas for now
- Frederic's OH is Tue 4:30 to 5:30 pm
- Haifeng's OH is Thur 4 to 5 pm

Announcements

A relater course from TTIC

TTIC 44000 - Special Topics: People, Society, and Algorithms

50 Units

This course considers designing and analyzing algorithms with a focus on explicitly taking consideration of people and society. The course covers selected topics in this area such as data elicitation, crowdsourcing, causal inference, etc., including recent research. The course will put an emphasis on theoretical principles underlying problems in these domains, including derivations and proofs of theoretical guarantees. Some application-specific considerations and directions will also be discussed as case studies. As this is an interdisciplinary field, we will also touch upon literature in psychology and economics that study the behavior of people.

Prerequisites: Knowledge of basic probability and linear algebra.

Topics include:

- Incentives: strictly proper scoring rules, Bayesian truth serum
- Crowdsourcing: learning from pairwise comparisons, crowdsourced labeling, parametric and non-parametric models and their relations, message-passing algorithms
- Causal inference: randomized controlled trials, experimental design, interference
- Fairness
- Applications: recommendation systems, peer review

DATA 37200: Learning, Decisions, and Limits
(Winter 2025)

Upper Confidence Bound (UCB) Algorithm

Instructor: Haifeng Xu

Outline

- First (Suboptimal) Attempt
- The UCB Algorithm
- Proving Regret Bound of UCB

Disclaimer

In this lecture, and likely many following ones…

We often ignore lower order terms and constant terms in our derivations, and use big O, Θ, Ω notations

- Mainly for clarity of argument, but all derivations are rigorous
- Very typical in computing research/analysis
- The argument is that once parameters are very large, lower order terms do not matter much ($\sqrt{T}$ vs $5\log T$)
- On technical side, it frees you from unimportant details and let you focus on major factors

Recap: Stochastic Multi-Armed Bandits (MAB)

$1: r_1 \sim D_1$

$2: r_2 \sim D_2$

$\cdots$

$k: r_k \sim D_k$

- A set of k arms, denoted as $[k] = \{1,2,\cdots,k\}$
- Pulling arm i once generates a random reward r_i drawn from σ-sub-Gaussian distribution D_i
- Algorithm designer plays for T rounds, and needs to decide which arm to pull to maximize your expected reward

Round	1	2	$\cdots$	t	$\cdots$	T
Algorithm's choice	i^1	i^2		i^t		i^T

Goal:

$$\max_{i^1,\cdots,i^T} \mathbb{E}\left[\sum_{t=1}^{T} r_{i^t}\right]$$

Recap: Stochastic Multi-Armed Bandits (MAB)

$1 : r_1 \sim D_1$

μ_1

$2 : r_2 \sim D_2$

μ_2

. . .

$k : r_k \sim D_k$

μ_k

Useful notations:

- let $\mu_i = \mathbb{E}[R_i]$, $\mu^* = \max_{i \in [k]} \mu_i$ and $i^* = \arg\max_{i \in [k]} \mu_i$ is the optimal arm
- Let $\Delta_i = \mu_i - \mu_{i^*}$ denote arm i's suboptimality gap
- Learner does not know μ_i's, D_i's, and Δ_i's
- For this lecture, assume learner knows T

A Natural First Attempt

$1: r_1 \sim D_1$

μ_1

$2: r_2 \sim D_2$

μ_2

. . .

$k: r_k \sim D_k$

μ_k

Q1: What is your most natural first attempt to solve this problem?

- Well, we want to find largest μ_i, but do not know it
- A natural idea is to learn the μ_i's to certain precision, and then pick the largest one
  - I.e., learning → then decisions (disentangled)
  - A well-known algorithm called *Explore Then Commit* (ETC)

A Natural First Attempt

$1: r_1 \sim D_1$

μ_1

$2: r_2 \sim D_2$

μ_2

. . .

$k: r_k \sim D_k$

μ_k

Q2: What's a natural algorithm to learn all μ_i's?

- D_i's are independent, and we want to learn its mean μ_i
- Can independently sample from D_i by pulling arm i repeatedly
- Nothing is known about μ_i, D_i, Δ_i

The most natural idea is to take n sample from each D_i, and use empirical mean as an estimation of μ_i!

A Natural First Attempt

$1: r_1 \sim D_1$

μ_1

$2: r_2 \sim D_2$

μ_2

. . .

$k: r_k \sim D_k$

μ_k

The *Explore Then Commit* (ETC) Algorithm

Algorithm parameter: n (satisfying $kn \leq T$)

1. (**Explore Phase**) For each arm i, pull it n times to draw n I.I.D. reward samples, and let $\bar{\mu}_i$ be the average of these n rewards

2. (**Commit Phase**) For round $t = kn + 1, \cdots, T$, pull arm $\bar{\imath} = \arg\max_{i \in [k]} \bar{\mu}_i$

Analysis of ETC

Challenge: needs to be smart about parameter n

- If too large, we may waste too much time learning in Step 1
- If too small, we may have large estimation error, hence commit to a very sub-optimal arm

The best n can be found by analyzing these two competing factors

The *Explore Then Commit* (ETC) Algorithm

Algorithm parameter: n (satisfying $kn \le T$)

1. (**Explore Phase**) For each arm i, pull it n times to draw n I.I.D. reward samples, and let $\bar{\mu}_i$ be the average of these n rewards

2. (**Commit Phase**) For round $t = kn + 1, \cdots, T$, pull arm $\bar{\imath} = \arg\max_{i \in [k]} \bar{\mu}_i$

Analysis of ETC

Challenge: needs to be smart about parameter n

- If too large, we may waste too much time learning in Step 1
- If too small, we may have large estimation error, hence commit to a very sub-optimal arm

The best n can be found by analyzing these two competing factors

1. Concentration inequality for σ-sub-Gaussian → estimation error as a function of #samples n

$$\Pr\left(\left|\frac{\sum_{i=1}^{n} r_i}{n} - \mu\right| \leq \sigma\sqrt{\frac{2\log T}{n}}\right) \geq 1 - 2/T^2$$

Analysis of ETC

Challenge: needs to be smart about parameter n

- If too large, we may waste too much time learning in Step 1
- If too small, we may have large estimation error, hence commit to a very sub-optimal arm

The best n can be found by analyzing these two competing factors

1. Concentration inequality for σ-sub-Gaussian → estimation error as a function of #samples n

$$\Pr\left(\left|\frac{\sum_{i=1}^{n} r_i}{n} - \mu\right| \leq \sigma\sqrt{\frac{2\log T}{n}}\right) \geq 1 - 2/T^2$$

2. Regret comes from two sources

<u>Regret from exploration</u> + <u>Regret from committing to suboptimal arm</u>

$$\sum_{i=1}^{k} \Delta_i n \quad + \quad 2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$$

Δ_i is the regret when exploring i

Analysis of ETC

Challenge: needs to be smart about parameter n
- If too large, we may waste too much time learning in Step 1
- If too small, we may have large estimation error, hence commit to a very sub-optimal arm

$\mu_{\bar{\imath}} + l_n \geq \bar{\mu}_{\bar{\imath}}$ where $l_n = \sigma\sqrt{\frac{2\log T}{n}}$ is the confidence length

$\geq \quad \bar{\mu}_{i^*}$ by our choice of $\bar{\imath}$ in Exploitation phase

$\geq \quad \mu_{i^*} - l_n$ with probability at least $1 - 4/T^2$ by union bound

Regret from exploration + Regret from committing to suboptimal arm

$\sum_{i=1}^{k} \Delta_i n$ $+$ $2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$

Δ_i is the regret when exploring i

Analysis of ETC

Challenge: needs to be smart about parameter n

- If too large, we may waste too much time learning in Step 1
- If too small, we may have large estimation error, hence commit to a very sub-optimal arm

$\mu_{\bar{\imath}} + l_n \geq \bar{\mu}_{\bar{\imath}}$ where $l_n = \sigma\sqrt{\frac{2\log T}{n}}$ is the confidence length

$\geq \quad \bar{\mu}_{i^*}$ by our choice of $\bar{\imath}$ in Exploitation phase

$\geq \quad \mu_{i^*} - l_n$ with probability at least $1 - 4/T^2$ by union bound

regret per round

Regret from exploration + Regret from committing to suboptimal arm

$$\sum_{i=1}^{k} \Delta_i n \quad + \quad 2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$$

Δ_i is the regret when exploring i

Analysis of ETC

To conclude the analysis:

with probability at least $1 - 4/T^2$, we have

$$\text{Regret}_T \le \sum_{i=1}^{k} \Delta_i n + 2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$$

$$\le Ckn + 2\sigma\sqrt{\frac{2\log T}{n}} \times T$$

- Assume all Δ_i's are upper bounded by constant C
- Will think of $T \gg k$ (otherwise, less interesting situations)

Analysis of ETC

To conclude the analysis:

with probability at least $1 - 4/T^2$, we have

$$\text{Regret}_T \le \sum_{i=1}^{k} \Delta_i n + 2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$$

$$\le Ckn + 2\sigma\sqrt{\frac{2\log T}{n}} \times T$$

$$\Rightarrow \text{Regret}_T \le \left[Ck + 2\sigma\sqrt{2\log T}\right] \times T^{2/3}$$ By letting $n = T^{2/3}$

$$= O\left(\left[k + \sqrt{\log T}\right] T^{\frac{2}{3}}\right)$$ Re-writing in Big-O notation

Remark:

- The choice of $n = T^{2/3}$ is not the exactly (though close to) the best, but it does achieve the best order of regret for ETC
- We use such tricks very often to trade exactness for cleaner analysis, without caring about constant value difference – beauty of big-O notation!

Analysis of ETC

To conclude the analysis:

with probability at least $\boldsymbol{1 - 4/T^2}$, we have

$$\text{Regret}_T \leq \sum_{i=1}^{k} \Delta_i n + 2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$$

$$\leq Ckn + 2\sigma\sqrt{\frac{2\log T}{n}} \times T$$

$$\Rightarrow \text{Regret}_T \leq \left[Ck + 2\sigma\sqrt{2\log T}\right] \times T^{2/3}$$ By letting $n = T^{2/3}$

$$= O\left(\left[k + \sqrt{\log T}\right] T^{\frac{2}{3}}\right)$$ Re-writing in Big-O notation

Analysis of ETC

To conclude the analysis:

with probability at least $1 - 4/T^2$, we have

$$\text{Regret}_T \leq \sum_{i=1}^{k} \Delta_i n \quad + \quad 2\sigma\sqrt{\frac{2\log T}{n}} \times (T - kn)$$

$$\leq \quad Ckn \; + 2\sigma\sqrt{\frac{2\log T}{n}} \times T$$

$$\Rightarrow \quad \text{Regret}_T \leq \left[Ck \; + 2\sigma\sqrt{2\log T}\right] \times T^{2/3}$$ By letting $n = T^{2/3}$

$$= O\left(\left[k \; + \sqrt{\log T}\right] T^{\frac{2}{3}}\right)$$ Re-writing in Big-O notation

with probability at most $4/T^2$

- We may have very bad luck, and picked a very bad arm suffering constant regret each round, leading to at most $\mathrm{T}C$ regret in total
- But accounting for its $\leq 4/T^2$ probability, this in expectation is $O(C)$ regret which is a constant

Analysis of ETC

Theorem: ETC algorithm suffers $O\left(\left[k + \sqrt{\log T}\right] T^{\frac{2}{3}}\right)$ regret for any MAB instance.

Analysis of ETC

Theorem: ETC algorithm suffers $O\left(\left[k + \sqrt{\log T}\right]T^{\frac{2}{3}}\right)$ regret for any MAB instance.

Question: Can we do better than such a disentangled algorithm that first learn (i.e., explore) and then decides/commits (i.e., exploit)?

Ans:

- Yes, but we need to blend exploration and exploitation together
- A representative and foundational algorithm is UCB that achieves regret

$$\text{Regret}_T = O\left(\sum_{i \neq i^*} \frac{\log T}{\Delta_i}\right)$$

- This bound depends on T logarithmically, and also depends on gaps Δ_i
  - hence called gap-dependent bound
  - though the UCB algorithm itself does not depend on knowledge of Δ_i's

Analysis of ETC

Theorem: ETC algorithm suffers $O\left(\left[k + \sqrt{\log T}\right]T^{\frac{2}{3}}\right)$ regret for any MAB instance.

Question: Can we do better than such a disentangled algorithm that first learn (i.e., explore) and then decides/commits (i.e., exploit)?

Ans:

- Yes, but we need to blend exploration and exploitation together
- A representative and foundational algorithm is UCB that achieves regret

$$\text{Regret}_T = O\left(\sum_{i \neq i^*} \frac{\log T}{\Delta_i}\right)$$

Can be converted to gap-independent bound $O(\sqrt{kT \log T})$

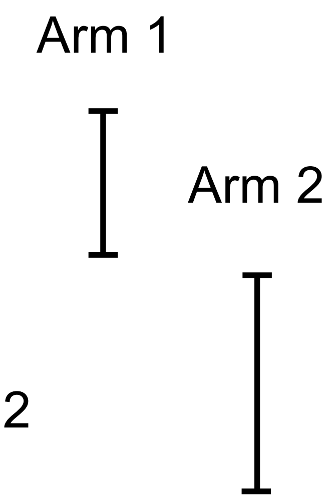
- This bound depends on T logarithmically, and also depends on gaps Δ_i
  - hence called gap-dependent bound
  - though the UCB algorithm itself does not depend on knowledge of Δ_i's

Outline

- First (Suboptimal) Attempt
- The UCB Algorithm
- Proving Regret Bound of UCB

In What Situations Is ETC Not Good?

- When one arm is significantly better than another
  - ETC was "too obsessed" with learning every arm accurately, and did not realize that we could have discarded obviously bad arms
  - (reflecting a key difference between learning decision and maximizing accuracy)
  - The *Upper Confidence Bound* (UBC) algorithm employs an elegant way to optimize this tradeoff



Once we detected this case, no need to waste further (highly sub-optimal) pulls to learn about arm 2

The Upper Confidence Bound (UCB) Algorithm

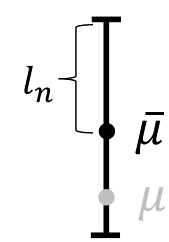
First things first, what is that (upper) confidence bound?

Comes from concentration inequality

- Given n sampled rewards $r_1, \cdots, r_n$ from any arm with mean μ and σ-sub-Gaussian reward distribution, we have

$$\Pr\left(|\bar{\mu} - \mu| \le \sigma\sqrt{\frac{\log 1/\delta}{n}}\right) \ge 1 - 2\delta \quad \text{where.} \quad \bar{\mu} = \frac{\sum_{i=1}^{n} r_i}{n}$$

- Denote $l_n = \sigma\sqrt{\frac{\log 1/\delta}{n}}$. So with probability at least $1 - 2\delta$, we have $\mu \in [\bar{\mu} - l_n, \bar{\mu} + l_n]$



The Upper Confidence Bound (UCB) Algorithm

First things first, what is that (upper) confidence bound?

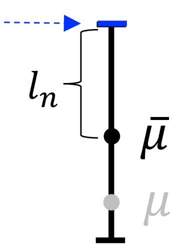
Comes from concentration inequality

- Given n sampled rewards $r_1, \cdots, r_n$ from any arm with mean μ and σ-sub-Gaussian reward distribution, we have

$$\Pr\left(|\bar{\mu} - \mu| \le \sigma \sqrt{\frac{\log 1/\delta}{n}} \right) \ge 1 - 2\delta \text{ where. } \bar{\mu} = \frac{\sum_{i=1}^{n} r_i}{n}$$

- Denote $l_n = \sigma \sqrt{\frac{\log 1/\delta}{n}}$. So with probability at least $1 - 2\delta$, we have $\mu \in [\bar{\mu} - l_n, \bar{\mu} + l_n]$

- $\bar{\mu} + l_n$ is called the upper confidence bound, which is fully calculable from sampled rewards
- Relatedly, $[\bar{\mu} - l_n, \bar{\mu} + l_n]$ is called the confidence interval of μ



The Upper Confidence Bound (UCB) Algorithm

First things first, what is that (upper) confidence bound?

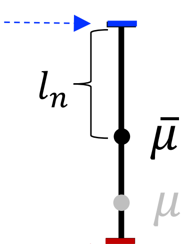
Comes from concentration inequality

- Given n sampled rewards $r_1, \cdots, r_n$ from any arm with mean μ and σ-sub-Gaussian reward distribution, we have

$$\Pr\left(|\bar{\mu} - \mu| \le \sigma\sqrt{\frac{\log 1/\delta}{n}}\right) \ge 1 - 2\delta \quad \text{where.} \quad \bar{\mu} = \frac{\sum_{i=1}^{n} r_i}{n}$$

- Denote $l_n = \sigma\sqrt{\frac{\log 1/\delta}{n}}$. So with probability at least $1 - 2\delta$, we have $\mu \in [\bar{\mu} - l_n, \bar{\mu} + l_n]$

- $\bar{\mu} + l_n$ is called the upper confidence bound, which is fully calculable from sampled rewards
- Relatedly, $[\bar{\mu} - l_n, \bar{\mu} + l_n]$ is called the confidence interval of μ



Guess what this is called?

Lower confidence bound

The Upper Confidence Bound (UCB) Algorithm

The *Upper Confidence Bound* (UCB) Algorithm

Parameter: δ

1. Initialization: $n_i = 0$ for each arm $i \in [n]$
2. For round $t = 1, 2 \cdots, T$

   2.1. pull the arm $i^t = \arg\max_{i \in [k]} \mathrm{ucb}_i(n_i : \delta)$ that has largest ucb (if any, ties are broken arbitrarily)

   3.2 update $n_{i^t} \leftarrow n_{i^t} + 1$, and update $\mathrm{ucb}_{i^t}(n_{i^t}; \delta)$

For any arm i, let n_i = #rounds arm i is pulled. Then define UCB as

$$\mathrm{ucb}_i(n_i; \delta) = \begin{cases} \infty, & n_i = 0 \\ \bar{\mu}_i + \sigma\sqrt{\dfrac{\log 1/\delta}{n_i}}, & n_i > 0 \end{cases}$$

The Upper Confidence Bound (UCB) Algorithm

The *Upper Confidence Bound* (UCB) Algorithm

Parameter: δ

1. Initialization: $n_i = 0$ for each arm $i \in [n]$
2. For round $t = 1, 2 \cdots, T$

   2.1. pull the arm $i^t = \arg\max_{i \in [k]} \mathrm{ucb}_i(n_i : \delta)$ that has largest ucb (if any, ties are broken arbitrarily)

   3.2 update $n_{i^t} \leftarrow n_{i^t} + 1$, and update $\mathrm{ucb}_{i^t}(n_{i^t}; \delta)$

Theorem: The regret of UCB with parameter $\delta = 1/T^2$ is upper bounded as follows:

$$\text{Regret} = O\left(\sum_{i \in [k], i \neq i^*} \frac{\log(T)}{\Delta_i}\right)$$

The Upper Confidence Bound (UCB) Algorithm

Remarks about UCB.

- The first k pulls will be arm $1, 2, \ldots, k$ has an arm with 0 pull has ∞ UCB
- In short, the algorithm uses UCB to guide choices and simply pick the one with largest UCB
  - Also called *optimism in the face of uncertainty* (OFU) principle
- Why UCB is a good idea for MAB
  - A large UCB must be due to either being pulled/exploited too little or large average reward

For any arm i, let n_i = #rounds arm i is pulled. Then define UCB as

$$\mathrm{ucb}_i(n_i; \delta) = \begin{cases} \infty, & n_i = 0 \\ \bar{\mu}_i + \sigma\sqrt{\frac{\log 1/\delta}{n_i}}, & n_i > 0 \end{cases}$$

The Upper Confidence Bound (UCB) Algorithm

Remarks about UCB.

- The first k pulls will be arm $1, 2, \dots, k$ has an arm with 0 pull has ∞ UCB
- In short, the algorithm uses UCB to guide choices and simply pick the one with largest UCB
  - Also called *optimism in the face of uncertainty* (OFU) principle
- Why UCB is a good idea for MAB
  - A large UCB must be due to either being pulled/exploited too little or large average reward
  - Hence UCB nicely blends exploration and exploitation together

For any arm i, let n_i = #rounds arm i is pulled. Then define UCB as

$$\mathrm{ucb}_i(n_i;\delta) = \begin{cases} \infty, & n_i = 0 \\ \bar{\mu}_i + \sigma\sqrt{\dfrac{\log 1/\delta}{n_i}}, & n_i > 0 \end{cases}$$

Outline

- First (Suboptimal) Attempt
- The UCB Algorithm
- Proving Regret Bound of UCB

Step 1: Understanding Where Regret Comes From

- Recall $u^* = \max_{i \in [k]} u_i$ is the mean of the best arm i^*
- Would have 0 regret if we always pulled i^*…so whenever we pulled some $i \neq i^*$ once, we suffer regret $\Delta_i = \mu^* - \mu_i$

Lemma 1 (Regret Decomposition): Let N_i denote the total number of times arm i is pulled by any algorithm for MAB. Then the algorithm's regret satisfies

$$\text{Regret} = \mathbb{E}\left[\sum_{i \in [k], i \neq i^*} \Delta_i N_i\right]$$

Proof: obvious from above explanations.

Challenges in remaining analysis – each N_i is a random variable, how do we upper bound its expected value?

Step 2: Identifying Good and Bad Events

- Randomness gives rise to a lot of situations/events – clearly, bad event may happen
  - E.g., with tiny probability, we may end up always pulling a bad arm

Core idea: we want to separately analyze good and bad events and hopefully show that bad events have very low probability

Step 2: Identifying Good and Bad Events

Definition: Define (random) good event E_t as "after pulling arm I_t at round t, arm I_t's true mean is within its confidence interval". That is,

$$E_t = \left\{ r_1, \cdots, r_{N_{I_t}} : \left| \bar{\mu}_{I_t} - \mu_{I_t} \right| \leq \sigma \sqrt{\frac{28 \log 1/\delta}{N_{I_t}}} \right\}$$

Lemma 2: for any t, $\Pr(E_t) \geq 1 - 2\delta$.

Caveats

- E_t is about $r_1, \cdots, r_{N_{I_t}}$ where N_{I_t} is a random variable
- Concentration inequality is only for a fixed number of samples n

I know this – concentration inequality!

Almost, but trickier, why?

Step 2: Identifying Good and Bad Events

Definition: Define (random) good event E_t as "after pulling arm I_t at round t, arm I_t's true mean is within its confidence interval". That is,

$$E_t = \left\{ r_1, \cdots, r_{N_{I_t}} : \left| \bar{\mu}_{I_t} - \mu_{I_t} \right| \le \sigma \sqrt{\frac{28 \log 1/\delta}{N_{I_t}}} \right\}$$

Lemma 2: for any t, $\Pr(E_t) \ge 1 - 2\delta$.

Proof.

- Conditioned on any realized i_t, n_{i_t}, $\{R_{i_t}^\tau - \mu_{i_t}^\tau\}_{\tau=1}^{n_{i_t}}$ is a martingale since given any history before τ, $\mathbb{E}\left(R_{i_t}^\tau - \mu_{i_t}^\tau\right) = 0$ always
- Azuma-Hoeffding inequality implies

$$\Pr\left(\left| \bar{r}_{i_t} - \mu_{i_t} \right| \le \sigma \sqrt{\frac{28 \log 1/\delta}{n_{i_t}}} \,\middle|\, i_t, n_{I_t} \right) \ge 1 - 2\delta$$

- Taking expectation w.r.t to i_t, i_{I_t} on both sides, we get $\Pr(E_t) \ge 1 - 2\delta$

Step 2: Identifying Good and Bad Events

Definition: Define (random) good event E_t as "after pulling arm I_t at round t, arm I_t's true mean is within its confidence interval". That is,

$$E_t = \left\{ r_1, \cdots, r_{N_{I_t}} : \left| \bar{\mu}_{I_t} - \mu_{I_t} \right| \leq \sigma \sqrt{\frac{28 \log 1/\delta}{N_{I_t}}} \right\}$$

Lemma 2: for any t, $\Pr(E_t) \geq 1 - 2\delta$.

Remark.

- Why we cannot apply standard Hoeffding inequality to $\{R_{i_t}^{\tau} - \mu_{i_t}^{\tau}\}_{\tau=1}^{n_{i_t}}$ C, after conditioning on i_t, n_{I_t}?
- Because conditioning on i_t, n_{I_t}, $\{R_{i_t}^{\tau}\}_{\tau=1}^{n_{i_t}}$ are not I.I.D. samples!
  - Be careful that some materials overlooked this subtle issue

Larger n_{i_t} → arm i_t is pulled more → realized past $R_{i_t}^{\tau}$'s are larger

Step 2: Identifying Good and Bad Events

Definition: Define (random) good event E_t as "after pulling arm I_t at round t, arm I_t's true mean is within its confidence interval". That is,

$$E_t = \left\{ r_1, \cdots, r_{N_{I_t}} : \left| \bar{\mu}_{I_t} - \mu_{I_t} \right| \le \sigma \sqrt{\frac{28 \log 1/\delta}{N_{I_t}}} \right\}$$

Lemma 3: $\Pr(\cap_{t=1}^{T} E_t) \ge 1 - 2T\delta$

Proof. Let $\overline{E_t}$ denotes complement of E_t, we have

$$\Pr(\cap_{t=1}^{n} E_t) = 1 - \Pr(\cup_{t=1}^{T} \overline{E_t})$$

$$\ge 1 - \sum_{t=1}^{T} \Pr(\overline{E_t})$$

Notably, this holds even when E_t's are correlated (and indeed they are)

Step 2: Identifying Good and Bad Events

Definition: Define (random) good event E_t as "after pulling arm I_t at round t, arm I_t's true mean is within its confidence interval". That is,

$$E_t = \left\{ r_1, \cdots, r_{N_{I_t}} : \left| \bar{\mu}_{I_t} - \mu_{I_t} \right| \le \sigma \sqrt{\frac{28 \log 1/\delta}{N_{I_t}}} \right\}$$

Lemma 3: $\Pr(\cap_{t=1}^{T} E_t) \ge 1 - 2T\delta$

Proof. Let $\overline{E_t}$ denotes complement of E_t, we have

$$\begin{aligned} \Pr(\cap_{t=1}^{n} E_t) &= 1 - \Pr(\cup_{t=1}^{T} \overline{E_t}) \\ &\ge 1 - \sum_{t=1}^{T} \Pr(\overline{E_t}) \\ &\ge 1 - 2T\delta \end{aligned}$$

Hence, setting $\delta = 1/T^2$, all good events simultaneously happen with probability $\ge 2/T$

Step 3: Bounding Regret under Good Events

- Now, we focus on situations where all E_t's happen, i.e., $\cap_{t=1}^{T} E_t$
- Since E_t is about the pulled arm I_t, and this is the only arm at round t whose confidence interval could possibly changes

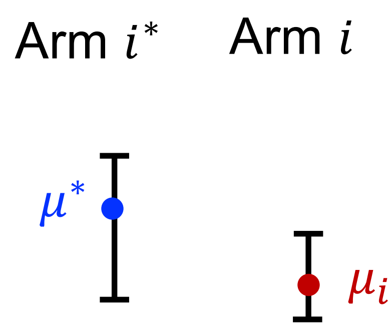
  → under $\cap_{t=1}^{T} E_t$, every arm i's mean is **always** within its confidence interval throughout the entire algorithm

Step 3: Bounding Regret under Good Events

Lemma 4: Under event $\cap_{t=1}^{T} E_t$, $\Pr\left(N_i \leq 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right) = 1$ for any $i \neq i^*$

Prove by contradiction:

- Suppose $N_i > 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1$, and let $N = \left\lfloor 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} \right\rfloor$
- We must have pulled arm i when its $N_i = N$
- Hence we have



$$
\begin{aligned}
\mathrm{ucb}_i(\mathrm{N}|\delta) &= \bar{\mu}_i + \sigma\sqrt{\frac{\log 1/\delta}{N}} \\
&\leq \bar{\mu}_i + \Delta_i/2 \\
&= \bar{\mu}_i - \Delta_i/2 + \Delta_i \\
&\leq \bar{\mu}_i - \sigma\sqrt{\frac{\log 1/\delta}{N}} + \Delta_i
\end{aligned}
$$

Plugging in $N \geq 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2}$

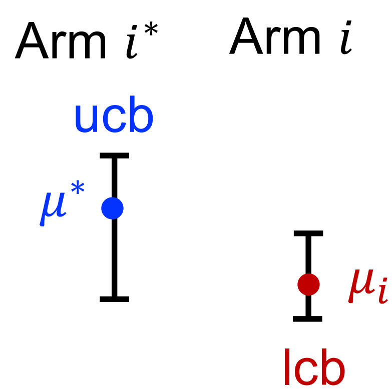
or equivalently, $\Delta_i \geq 2\sigma\sqrt{\frac{\log 1/\delta}{N}}$

Step 3: Bounding Regret under Good Events

Lemma 4: Under event $\cap_{t=1}^{T} E_t$, $\Pr\left(N_i \leq 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right) = 1$ for any $i \neq i^*$

Prove by contradiction:

- Suppose $N_i > 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1$, and let $N = \left\lceil 4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} \right\rceil$
- We must have pulled arm i when its $N_i = N$
- Hence we have



$$
\begin{aligned}
\mathrm{ucb}_i(\mathrm{N}|\delta) &= \bar{\mu}_i + \sigma\sqrt{\frac{\log 1/\delta}{N}} \\
&\leq \bar{\mu}_i - \sigma\sqrt{\frac{\log 1/\delta}{N}} + \Delta_i \\
&= \mathrm{lcb}_i(N|\delta) + \mu^* - \mu_i \\
&< \mu^* < ucb_{i^*}(N_{i^*}|\delta)
\end{aligned}
$$

So it is impossible that i's ucb can be larger than i^*'s, if $N_i = N$.

By definition of lower confidence bound and Δ_i

When in event $\cap_{t=1}^{T} E_t$

(Final) Step 4: Putting Everything Together

$$\text{Regret} = \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i\right]$$

By regret decomposition

$$= \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cap_{t=1}^{T} E_t\right] \times \Pr(\cap_{t=1}^{T} E_t)$$

$$+ \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cup_{t=1}^{T} \bar{E}_t\right] \times \Pr(\cup_{t=1}^{T} \bar{E}_t)$$

$$\leq \sum_{i\in[k],i\neq i^*} \Delta_i \times \left[4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right] + CT \times 2\delta T$$

By lemma 4, N_i is surely at most $\left[4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right]$

By lemma 3, the probably some bad event happens is at most $2\delta T$

(Final) Step 4: Putting Everything Together

$$\text{Regret} = \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i\right]$$

By regret decomposition

$$= \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cap_{t=1}^{T} E_t\right] \times \Pr(\cap_{t=1}^{T} E_t)$$

$$+ \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cup_{t=1}^{T} \bar{E}_t\right] \times \Pr(\cup_{t=1}^{T} \bar{E}_t)$$

$$\leq \sum_{i\in[k],i\neq i^*} \Delta_i \times \left[4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right] + CT \times 2\delta T$$

$$\leq \sum_{i\in[k],i\neq i^*} \left[8\sigma^2 \frac{\log(T)}{\Delta_i} + \Delta_i\right] + 2C$$

Plugging in $\delta = 1/T^2$

$$= O\left(\sum_{i\in[k],i\neq i^*} \frac{\log(T)}{\Delta_i}\right)$$

Computer science way to write it by using Big-O to hide all constants

This is called a gap-dependent regret bound (though running UCB does not need to know Δ_i's).

See issues? Very bad if some $\Delta_i \to 0$!

The Gap-Independent Regret Bound for UCB

$$\text{Regret} = \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i\right]$$

$$= \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cap_{t=1}^{T} E_t\right] \times \Pr(\cap_{t=1}^{T} E_t)$$

$$+ \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cup_{t=1}^{T} \bar{E}_t\right] \times \Pr(\cup_{t=1}^{T} \bar{E}_t)$$

$$\leq \sum_{i\in[k],i\neq i^*} \Delta_i \times \left[4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right] + CT \times 2\delta T$$

$$\min\left\{4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1, T\right\}$$

The Gap-Independent Regret Bound for UCB

$$\text{Regret} = \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i\right]$$

$$= \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cap_{t=1}^{T} E_t\right] \times \Pr(\cap_{t=1}^{T} E_t)$$

$$+ \mathbb{E}\left[\sum_{i\in[k],i\neq i^*} \Delta_i N_i \mid \cup_{t=1}^{T} \bar{E}_t\right] \times \Pr(\cup_{t=1}^{T} \bar{E}_t)$$

$$\leq \sum_{i\in[k],i\neq i^*} \Delta_i \times \left[4\sigma^2 \frac{\log(1/\delta)}{(\Delta_i)^2} + 1\right] + CT \times 2\delta T$$

$$= O\left(\sum_{i\in[k],i\neq i^*} \Delta_i \times \min\left\{\frac{\log(1/\delta)}{(\Delta_i)^2}, T\right\}\right)$$

$$= O\left(\sum_{i\in[k],i\neq i^*} \min\left\{\frac{\log(T)}{\Delta_i}, T\Delta_i\right\}\right)$$

$$= O\left(k\sqrt{T \log T}\right)$$

Caveat: there are tricks to refine last few steps to sharpen this bound to $O\left(\sqrt{k\, T \log T}\right)$;
Might be in homework ☺

Further Remarks

- There are other variants of UCB, some of which have slightly better bounds than this standard one we analyzed
  - However, all important ideas/techniques have been covered

- More generally, UCB is a kind of “index policy”
  - That is, designing an “index” to measure the value of each arm, and act purely based on this index
  - Such index policies are very useful, and find applications in many other cool problems such as Pandora’s box, restless bandits

Thank You

Haifeng Xu

University of Chicago

haifengxu@uchicago.edu