南方科技大学

MAT8034: Machine Learning

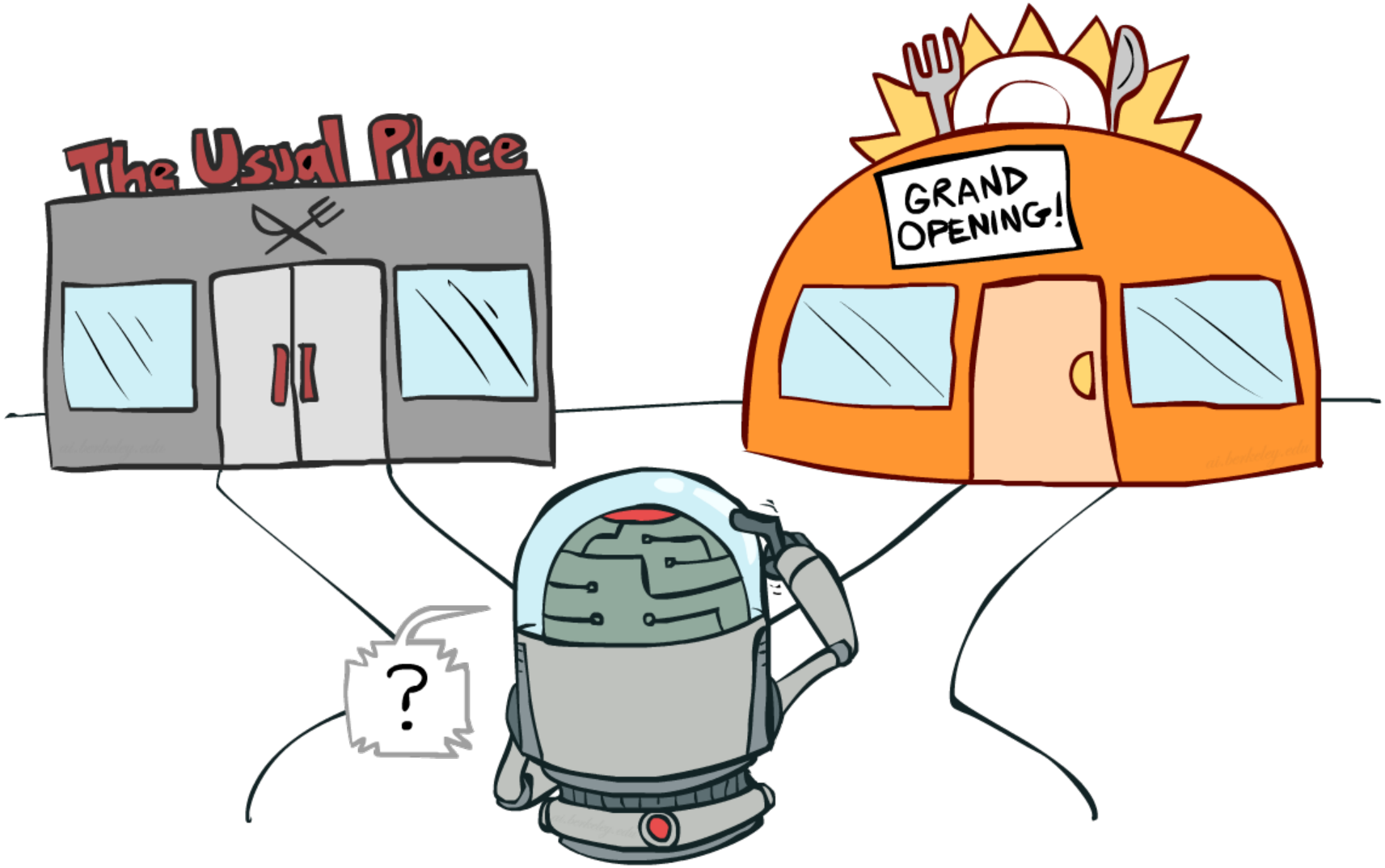
Exploration and Exploitation

Fang Kong

https://fangkongx.github.io/Teaching/MAT8034/Spring2026/index.html

Part of slide credits: ai.berkeley.edu; Emma Brunskill Stanford CS234

Exploration vs. Exploitation



Exploration vs. Exploitation

- ***Exploration***: try new things
- ***Exploitation***: do what's best given what you've learned so far
- Key point: pure exploitation often gets ***stuck in a rut*** and never finds an optimal policy!

Exploration method 1: ε-greedy

- ε-greedy exploration
  - Every time step, flip a biased coin
  - With (small) probability ε, act randomly
  - With (large) probability 1-ε, act on current policy

- Properties of ε-greedy exploration
  - Every s,a pair is tried infinitely often
  - Does a lot of stupid things
    - Jumping off a cliff *lots of times* to make sure it hurts
  - Keeps doing stupid things for ever
    - Decay ε towards 0

Demo Q-learning – Epsilon-Greedy – Crawler

Method 2: Optimistic Exploration Functions

- ***Exploration functions*** implement this tradeoff
  - Takes a value estimate u and a visit count n, and returns an optimistic utility, e.g., $f(u,n) = u + k/\sqrt{n}$

- Regular Q-update:
  - $Q(s,a) \leftarrow (1-\alpha) \cdot Q(s,a) + \alpha \cdot [R(s,a,s') + \gamma \max_a Q(s',a)]$
- Modified Q-update:
  - $Q(s,a) \leftarrow (1-\alpha) \cdot Q(s,a) + \alpha \cdot [R(s,a,s') + \gamma \max_a f(Q(s',a'),n(s',a'))]$
- Note: this propagates the "bonus" back to states that lead to unknown states as well!

Demo Q-learning – Exploration Function – Crawler

Evaluation criteria for RL algorithms

- How do we evaluate how “good” an algorithm is?
- If converges?
- If converges to optimal policy?
- How quickly reaches optimal policy?
- Mistakes make along the way?
- Will introduce common measures to evaluate RL algorithms

Multi-armed bandits

- Multi-armed bandit is a tuple of $(\mathcal{A}, \mathcal{R})$
- $\mathcal{A}$: known set of m actions (arms)
- $\mathcal{R}^a(r) = \mathbb{P}[r \mid a]$ is an unknown probability distribution over rewards
- At each step t the agent selects an action $a_t \in \mathcal{A}$
- The environment generates a reward $r_t \sim \mathcal{R}^{a_t}$
- Goal: Maximize cumulative reward $\sum_{\tau=1}^{t} r_\tau$

Greedy algorithm

- We consider algorithms that estimate $\hat{Q}_t(a) \approx Q(a) = \mathbb{E}[R(a)]$
- Estimate the value of each action by Monte-Carlo evaluation

$$\hat{Q}_t(a) = \frac{1}{N_t(a)} \sum_{i=1}^{t} r_i \mathbb{1}(a_i = a)$$

- The **greedy** algorithm selects the action with highest value

$$a_t^* = \arg\max_{a \in \mathcal{A}} \hat{Q}_{t-1}(a)$$

Greedy algorithm: an example

1. Sample each arm once
   - Take action a^1 ($r \sim$Bernoulli(0.95)), get 0, $\hat{Q}(a^1) = 0$
   - Take action a^2 ($r \sim$Bernoulli(0.90)), get $+1$, $\hat{Q}(a^2) = 1$
   - Take action a^3 ($r \sim$Bernoulli(0.1)), get 0, $\hat{Q}(a^3) = 0$
2. Will the greedy algorithm ever find the best arm in this case?

Greedy can lock onto suboptimal action

Assessing the performance of algorithms

- How do we evaluate the quality of a RL (or bandit) algorithm?
  - computational complexity, convergence, convergence to a fixed point, & empirical performance performance

- Now: introduce a formal measure of how well a RL/bandit algorithm will do in any environment, compared to optimal

Regret

- **Action-value** is the mean reward for action a

$$Q(a) = \mathbb{E}[r \mid a]$$

- **Optimal value** V^*

$$V^* = Q(a^*) = \max_{a \in \mathcal{A}} Q(a)$$

- **Regret** is the opportunity loss for one step, where the expectation is taken over the decision policy used to select a_t

$$l_t = \mathbb{E}[V^* - Q(a_t)]$$

Cumulative regret

- Over the horizon t, cumulative regret is the total opportunity loss

$$Reg_t = \mathbb{E}[\sum_{\tau=1}^{t} V^* - Q(a_\tau)]$$

- Maximize cumulative reward $\Leftrightarrow$ Minimize the cumulative regret

Regret decomposition

- **Count** $N_t(a)$ is number of times action a has been selected at time step t
- **Gap** Δ_a is the difference in value between action a and optimal action a^*, $\Delta_i = V^* - Q(a_i)$
- Regret is a function of gaps and counts

$$
\begin{aligned}
Reg_t &= \mathbb{E}\left[\sum_{\tau=1}^{t} V^* - Q(a_\tau)\right] \\
&= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)](V^* - Q(a)) \\
&= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)]\Delta_a
\end{aligned}
$$

Objective: small counts for large (unknown) gap

Hoeffding inequality

Lemma. (Hoeffding inequality) Let $Z_1, \ldots, Z_n$ be n independent and identically distributed (iid) random variables drawn from a Bernoulli(ϕ) distribution. I.e., $P(Z_i = 1) = \phi$, and $P(Z_i = 0) = 1 - \phi$. Let $\hat{\phi} = (1/n) \sum_{i=1}^{n} Z_i$ be the mean of these random variables, and let any $\gamma > 0$ be fixed. Then

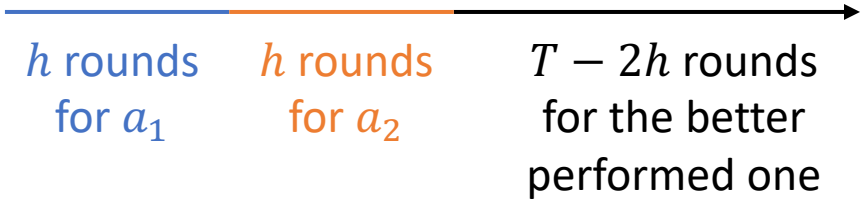
$$P(|\phi - \hat{\phi}| > \gamma) \leq 2 \exp(-2\gamma^2 n)$$

Explore-then-commit (ETC) [Garivier et al., 2016]

- There are $K = 2$ actions/arms
- Suppose
  - $Q(a_1) > Q(a_2)$
  - $\Delta = Q(a_1) - Q(a_2)$

A/B testing

- Explore-then-commit (ETC) algorithm
  - Select each arm h times
  - Find the empirically best arm A
  - Choose $A_t = A$ for all remaining rounds



Explore-then-commit (cont.)

- Regret analysis:

$$Reg(T) = T \cdot Q(a_1) - \mathbb{E}\left[\sum_{t=1}^{T} Q(a_t)\right]$$

Sample mean

$$= h\Delta + (T-2h) \cdot \Delta \cdot \mathbb{P}\left(\hat{Q}(a_1) < \hat{Q}(a_2)\right)$$

$$= h\Delta + (T-2h) \cdot \Delta \cdot \mathbb{P}\left(\left(\hat{Q}(a_2) - Q(a_2)\right) - \left(\hat{Q}(a_1) - Q(a_1)\right) > \Delta\right)$$

$$\leq \underbrace{h\Delta}_{\text{Exploration}} + \underbrace{T \cdot \Delta \cdot \exp\left(-\frac{h\Delta^2}{4}\right)}_{\text{Exploitation}}$$

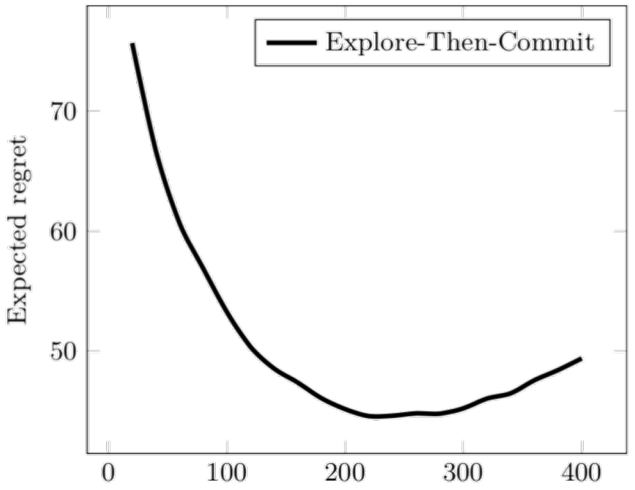
Hoeffding's inequality

$$\leq O\left(\frac{\log T}{\Delta}\right)$$

Choose $h = \left\lceil \frac{4}{\Delta^2} \log\left(\frac{T\Delta^2}{4}\right) \right\rceil$

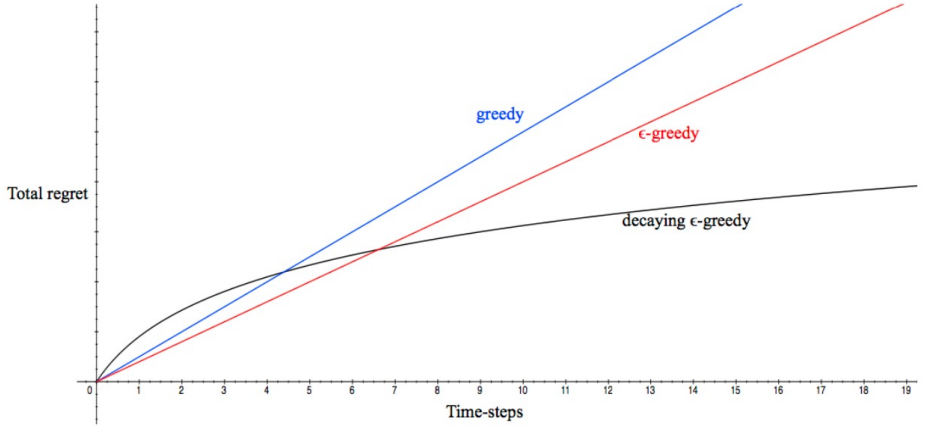
require the knowledge of Δ

- $Reg(T) = \Omega(T\Delta)$ if $h = 100$
- $Reg(T) = \Omega(T\Delta)$ if $h = T/10$



Only with the best choice of h the regret would be smallest

A soft version: ε-greedy



- For each round t
  - $\varepsilon_t \in (0,1)$
  - With probability ε_t, exploration (uniformly random select arms)
  - With probability $1 - \varepsilon_t$, exploitation (select the best performed arm so far)

- When $\varepsilon_t = \min\left\{1, \frac{c}{t\Delta^2}\right\}$, $Reg(T) = O\left(\frac{\log T}{\Delta}\right)$

Demo: https://cse442-17f.github.io/LinUCB/ Figure: CS234

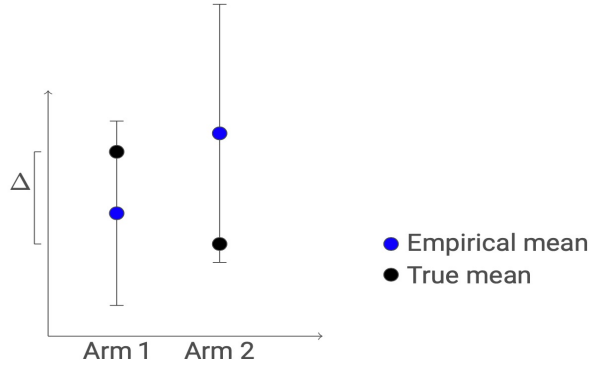
Upper confidence bound (UCB) [Auer et al., 2002]

- With high probability $\geq 1 - \delta$ By Hoeffding's inequality

$$Q(a_j) \in \left[\hat{Q}(a_j) - \sqrt{\frac{\log 1/\delta}{N_j}}, \hat{Q}(a_j) + \sqrt{\frac{\log 1/\delta}{N_j}}\right]$$

Sample mean

Number of selections of a_j



- Optimism: Believe arms have higher rewards, encourage exploration
  - The UCB value represents the reward estimates
- For each round t, select the arm

$$A(t) \in \text{argmax}_{j \in [K]} \left\{\hat{Q}(a_j) + \sqrt{\frac{\log 1/\delta}{N_j(t)}}\right\}$$

Upper confidence bound (UCB)

Exploitation

Exploration

Upper confidence bound (UCB) (cont.)

- Assume arm a_1 is the best arm
- If sub-optimal arm a_j is selected
  - w/ high probability

$$Q(a_1) \leq \text{UCB}_1 \leq \text{UCB}_j \leq Q(a_j) + 2\sqrt{\frac{\log 1/\delta}{N_j(t)}}$$

  - $\Rightarrow 2\sqrt{\frac{\log 1/\delta}{N_j(t)}} \geq \Delta_j := Q(a_1) - Q(a_j)$
  - $\Rightarrow N_j(t) \leq O\left(\frac{\log 1/\delta}{\Delta_j^2}\right)$

Can choose δ adaptive to time t

- By choosing $\delta = 1/T$, cumulative regret:

$$O\left(\sum_{j\neq 1} \frac{\log T}{\Delta_j^2} \cdot \Delta_j\right) = O(K \log T/\Delta)$$

$\Delta := \min_{j\neq 1}\Delta_j$
Without knowing Δ

- Assume arm a_1 is the best arm
- If sub-optimal arm a_j is selected
  - w/ high probability
    $Q(a_1) \leq \text{UCB}_1 \leq \text{UCB}_j \leq Q(a_j) + 2\sqrt{\frac{\log 1/\delta}{N_j(t)}}$
  - $\Rightarrow 2\sqrt{\frac{\log 1/\delta}{N_j(t)}} \geq \Delta_j := Q(a_1) - Q(a_j)$
  - $\Rightarrow N_j(t) \leq O\left(\frac{\log 1/\delta}{\Delta_j^2}\right)$
- By choosing $\delta = 1/T$, cumulative regret:
  $O\left(\sum_{j\neq 1} \frac{\log T}{\Delta_j^2} \cdot \Delta_j\right) = O(K \log T/\Delta)$

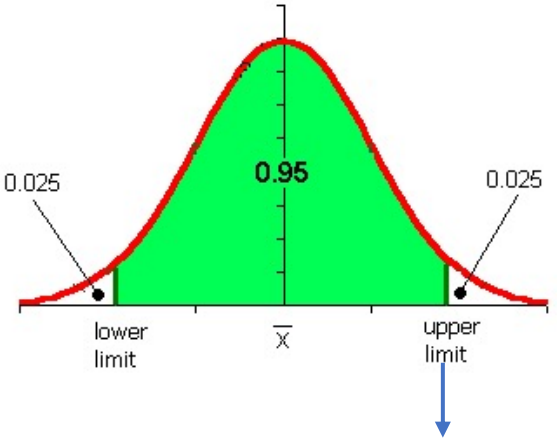
Thompson sampling (TS) [Agrawal and Goyal, 2013]

- Assume each arm has prior $\text{Gaussian}(0,1)$
- Sample an estimate $\tilde{Q}_j$ from the posterior distribution

$$\tilde{Q}_j \sim \text{Gaussian}\left(\hat{Q}_j, \frac{1}{1+N_j(t)}\right)$$

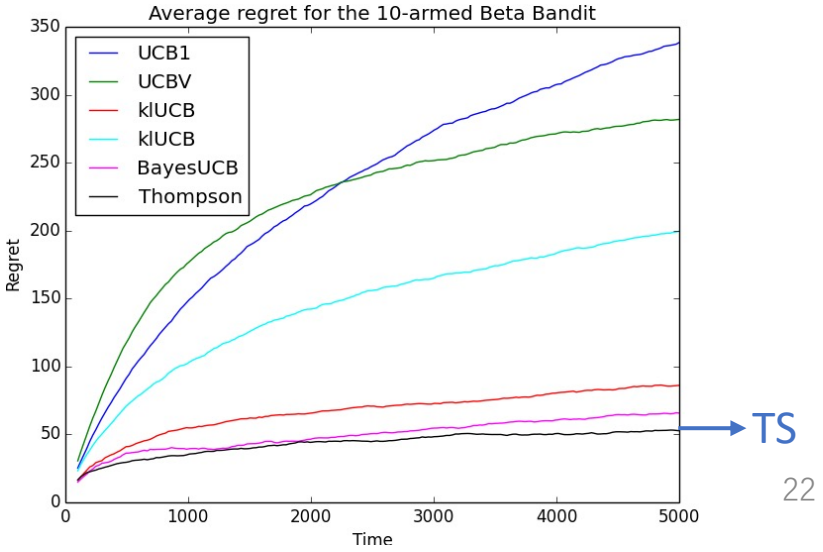
Exploitation

Exploration



- Select the arm $A(t) \in \text{argmax}_{j \in [K]} \tilde{Q}_j$

- Also have $O(K \log T/\Delta)$ regret
- Usually outperforms UCB



https://learnforeverlearn.com/bandits/

Regret type

- Problem(instance)-dependent: Bound regret as a function of the number of times we pull each arm and the gap between the reward for the pulled arm and the optimal arm
  - UCB, TS have instance-dependent regret of order $O(K\log T/\Delta)$

- Problem (instance)-independent: Bound how regret grows as a function of T
  - UCB, TS have instance-independent regret of order $\tilde{O}(\sqrt{KT})$

Summary

- Exploration in Q-learning
- Multi-armed bandit setting
- Regret definition, decomposition
- Algorithms
  - ETC, epsilon-greedy, UCB, TS

References I

- Lattimore, Tor, and Csaba Szepesvári. Bandit algorithms. Cambridge University Press, 2020.
- Garivier, Aurélien, Tor Lattimore, and Emilie Kaufmann. "On explore-then-commit strategies." Advances in Neural Information Processing Systems 29 (2016).
- Audibert, Jean-Yves, and Sébastien Bubeck. "Best arm identification in multi-armed bandits." COLT-23th Conference on learning theory-2010. 2010.
- Auer, Peter, Nicolo Cesa-Bianchi, and Paul Fischer. "Finite-time analysis of the multiarmed bandit problem." Machine learning 47 (2002): 235-256.
- Agrawal, Shipra, and Navin Goyal. "Further Optimal Regret Bounds For Thompson Sampling." Sixteenth International Conference on Artificial Intelligence and Statistics. 2013.