Contextual Bandits

Lucas Janson **CS/Stat 184(0): Introduction to Reinforcement Learning Fall 2024**

- Feedback from last lecture
- Recap
- UCB-VI for linear MDPs
- Recall: Contextual Bandits
- LinUCB



Feedback from feedback forms

1. Thank you to everyone who filled out the forms!





- Recap
- UCB-VI for linear MDPs
- Recall: Contextual Bandits
- LinUCB

Exploration in MDP: make it a bandit and do UCB?

Q: given a discrete MDP, how many unique deterministic policies are there?

This seems bad, so are MDPs just super hard or can we do better?

 $\left(|A|^{|S|} \right)^{H}$

So treating each policy as an "arm" and running UCB gives us regret $\tilde{O}(\sqrt{|A|^{|S|H}N})$



Tabular UCB-VI

For $n = 1 \rightarrow N$: 1. Set $N_h^n(s, a) = \sum_{k=1}^{n-1} \mathbf{1}\{(s_h^i, a_h^i) = (s, a)\}, \forall s, a, h$ $i = 1_{n-1}$ 2. Set $N_h^n(s, a, s') = \sum \mathbf{1}\{(s_h^i, a_h^i, s_{h+1}^i) = (s, a, s')\}, \forall s, a, a', h$ i=13. Estimate \hat{P}^n : $\hat{P}^n_h(s' \mid s, a) = \frac{N^n_h(s, a, s)}{N^n_h(s, a)}$ 4. Plan: $\pi^{n} = VI\left(\{\hat{P}_{h}^{n}, r_{h} + b_{h}^{n}\}_{h}\right)$, with 5. Execute π^n : { $s_0^n, a_0^n, r_0^n, \dots, s_{H-1}^n, a_{H-1}^n, r_{H-1}^n, s_H^n$ }

$$\frac{s')}{p}, \forall s, a, s', h$$

$$b_h^n(s, a) = cH \sqrt{\frac{\log(|S||A|HN/\delta)}{N_h^n(s, a)}}$$

High-level Idea: Exploration Exploitation Tradeoff

Upper bound per-episode regret: $V_0^{\star}(s_0) - V_0^{\pi^n}(s_0) \leq \hat{V}_0^n(s_0) - V_0^{\pi^n}(s_0)$ by construction of b_h^n

2. What if $\hat{V}_0^n(s_0) - V_0^{\pi^n}(s_0)$ is large?

Some $b_h^n(s, a)$ must be large (or some $\hat{P}_h^n(\cdot | s, a)$ estimation errors must be large, but with high probability any $\hat{P}_{h}^{n}(\cdot | s, a)$ with high error must have small $N_{h}^{n}(s, a)$ and hence high $b_{h}^{n}(s, a)$

$$\mathbb{E}\left[\mathsf{Regret}_{N}\right] := \mathbb{E}\left[\sum_{n=1}^{N} \left(V^{\star} - V^{\pi^{n}}\right)\right] \leq \widetilde{O}\left(H^{2}\sqrt{|S||A|N}\right)$$

1. What if $\hat{V}_0^n(s_0) - V_0^{\pi^n}(s_0)$ is small?

Then π^n is close to π^* , i.e., we are doing <u>exploitation</u>

Large $b_h^n(s, a)$ means π^n is being encouraged to do (s, a), since it will apparently have very high reward, i.e., exploration









- Recap
 - UCB-VI for linear MDPs
 - Recall: Contextual Bandits
 - LinUCB



Linear MDP Definition

S & A could be large or even continuous, hence poly(|S|, |A|) is not acceptable

$$P_{h}(s'|s,a) = \mu_{h}^{\star}(s') \cdot \phi(s,a), \quad \mu_{h}^{\star} : S \mapsto \mathbb{R}^{d}, \quad \phi : S \times A \mapsto \mathbb{R}^{d}$$
$$r(s,a) = \theta_{h}^{\star} \cdot \phi(s,a), \quad \theta_{h}^{\star} \in \mathbb{R}^{d}$$

Finite horizon time-dependent episodic MDP $\mathcal{M} = \{S, A, H, \{r\}_h, \{P\}_h, s_0\}$

Feature map ϕ is known to the learner! (We assume reward is known, i.e., θ^{\star} is known)

Planning in Linear MDP: Value Iteration $P_h(\cdot | s, a) = \mu_h^* \phi(s, a), \quad \mu_h^* \in \mathbb{R}^{|S| \times d}, \quad \phi(s, a) \in \mathbb{R}^d$ $r_h(s, a) = (\theta_h^*)^\top \phi(s, a), \quad \theta_h^* \in \mathbb{R}^d$

$$V_{H}^{\star}(s) = 0, \forall s,$$

$$Q_{h}^{\star}(s, a) = r_{h}(s, a) + \mathbb{E}_{s' \sim P_{h}(\cdot | s, a)} V_{h+1}^{\star}(s')$$

$$= \theta_{h}^{\star} \cdot \phi(s, a) + (\mu_{h}^{\star} \phi(s, a))^{\top} V_{h+1}^{\star}$$

$$= \phi(s, a)^{\top} (\theta_{h}^{\star} + (\mu_{h}^{\star})^{\top} V_{h+1}^{\star})$$

$$= \phi(s, a)^{\top} w_{h}$$

$$T_{h}^{\star}(s) = \max \phi(s, a)^{\top} w_{h}, \quad \pi_{h}^{\star}(s) = \arg \max \phi(s, a)^{\top} w_{h}$$

Indeed we can show that $Q_h^{\pi}(\cdot, \cdot)$ Is linear with respect to ϕ as well, for any π, h

a

At the beginning of iteration n:

1. Learn transition model $\{\hat{P}_{h}^{n}\}_{h=0}^{H-1}$ from all previous data $\{s_{h}^{i}, a_{h}^{i}, s_{h+1}^{i}\}_{i=0}^{n-1}$

3. Plan: $\pi^{n+1} =$

UCBVI in Linear MDPs

2. Design reward bonus $b_h^n(s, a), \forall s, a$

$$\mathsf{VI}\left(\{\hat{P}^n\}_h,\{r_h+b_h^n\}\right)$$

How to estimate $\{\hat{P}_{h}^{n}\}_{h=0}^{H-1}$?

Given s, a, note that $\mathbb{E}_{s' \sim P_{h}(\cdot | s, a)}$

Penalized Linear Regression:

$$\min_{\mu} \sum_{i=1}^{n-1} \|\mu \phi(s_h^i, a_h^i) - \delta(s_{h+1}^i)\|_2^2 + \lambda \|\mu\|_F^2$$

$$A_{h}^{n} = \sum_{i=1}^{n-1} \phi(s_{h}^{i}, a_{h}^{i}) \phi(s_{h}^{i}, a_{h}^{i})^{\mathsf{T}} + \lambda I$$

 $\hat{P}_h^n(\cdot \mid s, a) = \hat{\mu}_h^n \phi(s, a)$

Denote $\delta(s) \in \mathbb{R}^{|S|}$ with zero everywhere except the entry corresponding to s

$$_{a)}\left[\delta(s')\right] = P_{h}(\cdot \mid s, a) = \mu_{h}^{\star}\phi(s, a)$$

$$\widehat{u}_{h}^{n} = (A_{h}^{n})^{-1} \sum_{i=1}^{n-1} \delta(s_{h+1}^{i}) \phi(s_{h}^{i}, a_{h}^{i})^{\mathsf{T}}$$

Chebyshev-like approach, similar to in linUCB (will cover later this lecture):

How to choose $b_h^n(s, a)$?

 $b_h^n(s,a) = \beta \sqrt{\phi(s,a)^{\mathsf{T}}(A_h^n)^{-1}\phi(s,a)}, \quad \beta = \widetilde{O}(dH)$

linUCB-VI: Put All Together

For $n = 1 \rightarrow N$: 1. Set $A_h^n = \sum_{k=1}^{n-1} \phi(s_h^i, a_h^i) \phi(s_h^i, a_h^i)^{\top} + \lambda I$ $i=1 \qquad n-1 \\ \text{2. Set } \hat{\mu}_h^n = (A_h^n)^{-1} \sum_{k=1}^{n-1} \delta(s_{h+1}^i) \phi(s_h^i, a_h^i)^\top$ i=1

3. Estimate \hat{P}^n : $\hat{P}^n_h(\cdot | s, a) = \hat{\mu}^n_h \phi(s, a)$

4. Plan: $\pi^n = \text{VI}\left(\{\hat{P}_h^n, r_h + b_h^n\}_h\right)$, with $b_h^n(s, a) = cdH_{\sqrt{\phi(s, a)^T(A_h^n)^{-1}\phi(s, a)}}$

5. Execute

$$\mathbb{E}\left[\mathsf{Regret}_{N}^{n}, a_{0}^{n}, r_{0}^{n}, \dots, s_{H-1}^{n}, a_{H-1}^{n}, r_{H-1}^{n}, s_{H}^{n}\right] \\ \mathbb{E}\left[\mathsf{Regret}_{N}^{n}\right] := \mathbb{E}\left[\sum_{n=1}^{N} \left(V^{\star} - V^{\pi^{n}}\right)\right] \leq \widetilde{O}\left(H^{2}d^{1.5}\sqrt{N}\right)$$

No *S*, *A* dependence!





- Recap
- UCB-VI for linear MDPs
 - Recall: Contextual Bandits
 - LinUCB



Recall: (non-contextual) bandit

- Each arm has an <u>unknown</u> reward distribution, i.e., $\nu_k \in \Delta([0,1])$, w/ mean $\mu_k = \mathbb{E}_{r \sim \nu_k}[r]$

For
$$t = 0 \rightarrow T - 1$$

- 1. Learner pulls arm $a_t \in \{1, \dots, K\}$
- 2. Learner observes an i.i.d reward $r_t \sim \nu_{a_t}$ of arm a_t

$$\text{Regret}_{T} = T\mu^{\star} - \sum_{t=0}^{T-1} \mu_{a_{t}} = \sum_{t=0}^{T-1} (\mu^{\star} - \mu_{a_{t}})$$

We have K many arms; label them $1, \ldots, K$

(based on historical information)

Recall: Beyond simple bandits

on all websites:

- maybe some types of users prefer one ad while others prefer another, or maybe one type of ad works better on certain websites while another
- works better on other websites

Which user comes in next is random, but we have some context to tell situations apart and hence learn different optimal actions

In a bandit, we are presented with the same decision at every time In practice, often decisions are not the same every time

E.g., in online advertising there may not be a single best ad to show all users

Recall: Contextual bandit environment

- Context at time t encoded into a variable x_t that we see before choosing our action x_t is drawn i.i.d. at each time point from a distribution ν_x on sample space \mathscr{X}
- x_t then affects the reward distributions of each arm, i.e., if we choose arm k, we get a reward that is drawn from a distribution that depends on x_t , namely, $\nu^{(k)}(x_t)$
 - Accordingly, we should also choose our action a_t in a way that depends on x_t , i.e., our action should be chosen by a function of x_t (a policy), namely, $\pi_t(x_t)$
 - If we knew everything about the environment, we'd want to use the optimal policy $\pi^{\star}(x_t) := \arg \max_{k \in \{1, \dots, K\}} \mu^{(k)}(x_t),$ where $\mu^{(k)}(x) := \mathbb{E}_{r \sim \nu^{(k)}(x)}[r]$

 π^{\star} is the policy we compare to in computing regret







Recall: Contextual bandit environment

For
$$t = 0 \rightarrow T - 1$$

- 2. Learner pulls arm $a_t = \pi_t(x_t) \in \{1, \dots, K\}$

Note that if the context distribution ν_{χ} always returns the same value (e.g., 0), then the contextual bandit <u>reduces</u> to the original multi-armed bandit

Formally, a contextual bandit is the following interactive learning process:

1. Learner sees context $x_t \sim \nu_x$ Independent of any previous data π_{t} policy learned from all data seen so far 3. Learner observes reward $r_t \sim \nu^{(a_t)}(x_t)$ from arm a_t in context x_t







Recall: UCB for contextual bandits

- mean and number of arm pulls separately for each value of the context all arm mean estimates $\hat{\mu}_{t}^{(k)}(x)$, of which there are $K|\mathcal{X}|$ instead of just K
- Added x_t argument to $\hat{\mu}_t^{(k)}$ and $N_t^{(k)}$ since we now keep track of the sample - Added $|\mathcal{X}|$ inside the log because our union bound argument is now over

- UCB algorithm conceptually identical as long as $|\mathcal{X}|$ finite:
 - $\pi_t(x_t) = \arg\max_k \hat{\mu}_t^{(k)}(x_t) + \sqrt{\ln(2TK|\mathcal{X}|/\delta)/2N_t^{(k)}(x_t)}$

- But when $|\mathcal{X}|$ is really big (or even infinite), this will be really bad!
- <u>Solution</u>: share information across contexts x_t , i.e., <u>don't</u> treat $\nu^{(k)}(x)$ and $\nu^{(k)}(x')$ as completely different distributions which have nothing to do with one another Example: showing an ad on a NYT article on politics vs a NYT article on sports:
 - Not *identical* readership, but still both on NYT, so probably still similar readership!



Recall: Modeling in contextual bandits

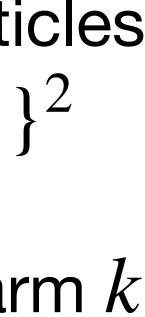
Need a model for $\mu^{(k)}(x)$, e.g., a linear model: $\mu^{(k)}(x) = \theta_k^{\mathsf{T}} x$

E.g., placing ads on NYT or WSJ (encoded as 0 or 1 in the first entry of x), for articles on politics or sports (encoded as 0 or 1 in the second entry of x) $\Rightarrow x \in \{0,1\}^2$

Lower dimension makes learning easier, but model could be wrong/biased

 $|\mathcal{X}| = 4 \Rightarrow$ w/o linear model, need to learn 4 different $\mu^{(k)}(x)$ values for each arm k

With linear model there are just 2 parameters: the two entries of $\theta_k \in \mathbb{R}^2$





- Recap
- UCB-VI for linear MDPs
- Recall: Contextual Bandits
 - LinUCB



Linear model fitting

How to estimate $\theta^{(k)}$? Linear regression

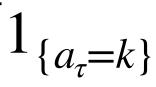
Least squares estimator: $\hat{\theta}_{t}^{(k)}$

Minimize squared error over time points when arm k selected

$$\begin{aligned} \text{Claim:} \ \hat{\theta}_{t}^{(k)} &= \left(\sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}}\right)^{-1} \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} \\ \text{proof:} \ \nabla_{\theta} \left[\sum_{\tau=0}^{t-1} (r_{\tau} - x_{\tau}^{\top} \theta)^{2} \mathbf{1}_{\{a_{\tau}=k\}}\right] &= 2 \sum_{\tau=0}^{t-1} x_{\tau} (r_{\tau} - x_{\tau}^{\top} \theta) \mathbf{1}_{\{a_{\tau}=k\}} = 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} \\ &= 0 \quad \Rightarrow \quad \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}}$$

Linear model for rewards: $\mu^{(k)}(x) = x^{\mathsf{T}}\theta^{(k)}$

$$= \arg\min_{\theta \in \mathbb{R}^d} \sum_{\tau=0}^{t-1} (r_{\tau} - x_{\tau}^{\mathsf{T}}\theta)^2 \mathbf{1}_{\{a_{\tau}=k\}}$$



Linear model fitting (cont'd)

$$\begin{aligned} \text{Recall:} \ \hat{\theta}_{t}^{(k)} &= \left(\sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}}\right)^{-1} \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} \\ \text{Let} \ A_{t}^{(k)} &= \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \text{ and } b_{t}^{(k)} &= \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} \\ \text{Then} \ \hat{\theta}_{t}^{(k)} &= \left(A_{t}^{(k)}\right)^{-1} b_{t}^{(k)} \end{aligned}$$

 $A_t^{(k)}$ like empirical covariance matrix of the contexts when arm k was chosen $b_{\star}^{(k)}$ like empirical covariance between contexts and rewards when arm k was chosen $A_{t}^{(k)}$ must be invertible, which basically requires $N_{t}^{(k)} \geq d$







Uncertainty quantification

- For UCB, recall that we need <u>confidence bounds</u> on the expected reward of each arm (given context x_t)
- Hoeffding was the main tool so far, but it used the fact that our estimate for the expected reward was a sample mean of the rewards we'd seen so far in the same setting (action, context)
- With a model, we can use rewards we've seen in other settings \rightarrow better estimation
 - But not using sample mean as estimator, so need something other than Hoeffding
 - <u>Chebyshev's inequality</u>: for a mean-zero random variable Y,
 - $|Y| \le \beta \sqrt{\mathbb{E}[Y^2]}$ with probability $\ge 1 1/\beta^2$



Uncertainty quantification (cont'd)

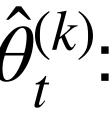
- Want confidence bounds on our estimated mean rewards for each arm: $x_t^{\top} \hat{\theta}_{\star}^{(k)}$
 - Strategy: apply Chebyshev's inequality to $x_t^T \hat{\theta}_t^{(k)} x_t^T \theta^{(k)}$
 - Need: $\mathbb{E}[x_t^{\mathsf{T}}\hat{\theta}_t^{(k)} x_t^{\mathsf{T}}\theta^{(k)}]$ (make sure it's zero) and $\mathbb{E}[x_t^{\mathsf{T}}\hat{\theta}_t^{(k)} x_t^{\mathsf{T}}\theta^{(k)}]^2$

Let
$$w_t = r_t - \mathbb{E}_{r \sim \nu^{(k)}(x_t)}[r] = r_t - x_t^{\mathsf{T}} \theta^{(k)}$$

$$\hat{\theta}_{t}^{(k)} = (A_{t}^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_{\tau} r_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = (A_{t}^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_{\tau} (x_{\tau}^{\top} \theta^{(k)} + w_{\tau}) \mathbf{1}_{\{a_{\tau}=k\}}$$
$$= (A_{t}^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} \mathbf{1}_{\{a_{\tau}=k\}} \theta^{(k)} + (A_{t}^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_{\tau} w_{\tau} \mathbf{1}_{\{a_{\tau}=k\}} = \theta^{(k)} + (A_{t}^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_{\tau} \mathbf{1}_{\{a_{\tau}=k\}}$$

^{k)}, and we derive a useful expression for $\hat{\theta}_{t}^{(k)}$:







Uncertainty quantification (cont'd)

Recall:
$$\hat{\theta}_t^{(k)} = \theta^{(k)} + (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_\tau \mathbf{1}_{\{a_\tau=k\}} w_\tau$$

Assume for simplicity that we are doing pure exploration, so the actions at each time step are totally independent of everything else.

$$\begin{split} \mathbb{E}_{w_{\tau}}[x_{t}^{\top}\hat{\theta}_{t}^{(k)} - x_{t}^{\top}\theta^{(k)}] &= \mathbb{E}_{w_{\tau}}[x_{t}^{\top}(A_{t}^{(k)})^{-1}\sum_{\tau=0}^{t-1}x_{\tau}\mathbf{1}_{\{a_{\tau}=k\}}w_{\tau}] = x_{t}^{\top}(A_{t}^{(k)})^{-1}\sum_{\tau=0}^{t-1}x_{\tau}\mathbf{1}_{\{a_{\tau}=k\}}\mathbb{E}_{w_{\tau}}[w_{\tau}] = \\ \mathbb{E}_{w_{\tau}}[(x_{t}^{\top}\hat{\theta}_{t}^{(k)} - x_{t}^{\top}\theta^{(k)})^{2}] &= \mathbb{E}_{w_{\tau}}\left[\left(x_{t}^{\top}(A_{t}^{(k)})^{-1}\sum_{\tau=0}^{t-1}x_{\tau}\mathbf{1}_{\{a_{\tau}=k\}}w_{\tau}\right)^{2}\right] \\ &= x_{t}^{\top}(A_{t}^{(k)})^{-1}\sum_{\tau=0}^{t-1}\sum_{\tau'=0}^{t-1}x_{\tau}x_{\tau'}^{\top}\mathbf{1}_{\{a_{\tau}=k\}}\mathbb{E}_{w_{\tau}}\left[w_{\tau}w_{\tau'}\right](A_{t}^{(k)})^{-1}x_{t} \\ &= x_{t}^{\top}(A_{t}^{(k)})^{-1}\sum_{\tau=0}^{t-1}x_{\tau}x_{\tau}^{\top}\mathbf{1}_{\{a_{\tau}=k\}}\mathbb{E}_{w_{\tau}}[w_{\tau}^{2}](A_{t}^{(k)})^{-1}x_{t} \leq x_{t}^{\top}(A_{t}^{(k)})^{-1}A_{t}^{(k)}(A_{t}^{(k)})^{-1}x_{t} = x_{t}^{\top}(A_{t}^{(k)})^{-1}x_{t} \end{split}$$





Chebyshev confidence bounds + intuition Chebyshev: $x_t^{\top} \theta^{(k)} \leq x_t^{\top} \hat{\theta}_t^{(k)} + \beta \sqrt{x_t^{\top} (A_t^{(k)})^{-1} x_t}$ with probability $\geq 1 - 1/\beta^2$ $A_t^{(k)} = \sum_{\tau}^{t-1} x_{\tau} x_{\tau}^{\top} 1_{\{a_{\tau} = k\}}$

Intuition:

UCB term 1: $x_t^T \hat{\theta}^{(k)}$ large when context and coefficient estimate aligned UCB term 2: $x_t^{\mathsf{T}}(A_t^{(k)})^{-1}x_t = \frac{1}{N_t^{(k)}}x_t^{\mathsf{T}}(\Sigma_t^{(k)})^{-1}x_t$, where $\Sigma_{t}^{(k)} = \frac{1}{N_{t}^{(k)}} A_{t}^{(k)} = \frac{1}{N_{t}^{(k)}} \sum_{\tau=0}^{t-1} x_{\tau} x_{\tau}^{\top} 1_{\{a_{\tau}=k\}} \text{ is the empirical covariance}$ matrix of contexts when arm k chosen

Large when $N_t^{(k)}$ small or x_t not aligned with historical data



Some issues

Issue 1: All this assumed pure exploration!

Issue 2: $A_{\star}^{(k)}$ has to be invertible

Solution (to both issues): regularize

- Recall from HW 1 that we don't even expect unbiasedness for our arm mean estimates in the simple bandit case, due to adaptivity
 - So actually, the bounds we got don't really apply...

Before the *d*th time that arm *k* gets pulled, $\hat{\theta}_{t}^{(k)}$ undefined

- Replace $A_{t}^{(k)} \leftarrow A_{t}^{(k)} + \lambda I$ for some $\lambda > 0$
- Makes $A_{r}^{(k)}$ invertible always, and it turns out a bound just like Chebyshev's applies (with more details and a much more complicated proof, which we won't get into)



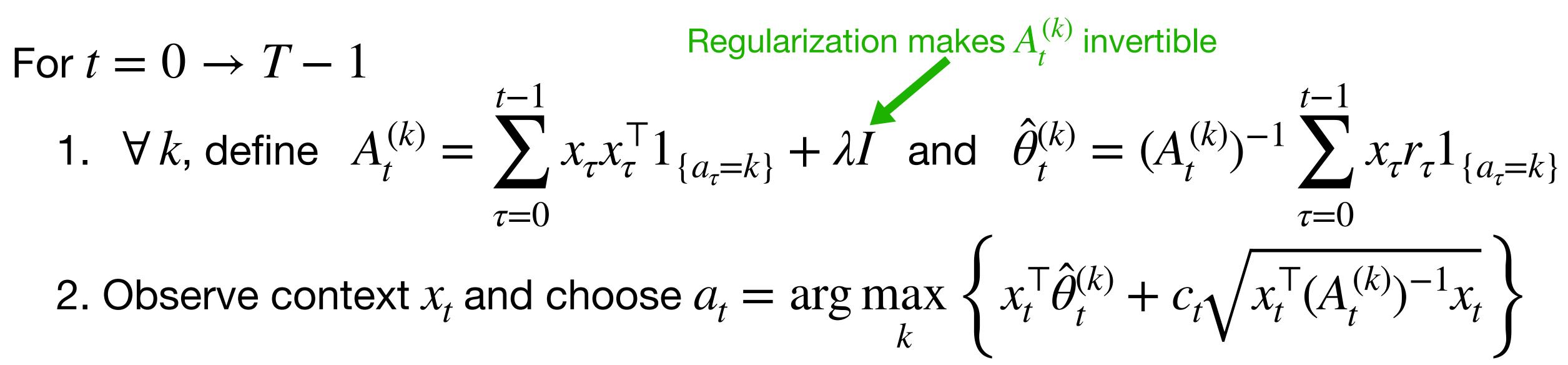
LinUCB algorithm

 $\tau = 0$

3. Observe reward $r_t \sim \nu^{(a_t)}(x_t)$

 c_t similar to log term in (non-lin)UCB, in that it depends logarithmically on i. $1/\delta$ (δ is probability you want the bound to hold with) ii. t and d implicitly via $det(A_t^{(k)})$

Can prove \tilde{O} (



$$(\sqrt{T})$$
 regret bound

Extensions

- 1. Can always replace contexts x_t with any fixed (vector-valued) function $\phi(x_t)$ E.g., if believe rewards quadratic in scalar x_t , could make $\phi(x_t) = (x_t, x_t^2)$
- 2. Instead of fitting different $\theta^{(k)}$ for each arm, we could assume the mean reward is linear in some function of both the context and the action, i.e., $\mathbb{E}_{r \sim \nu^{a_t(x_t)}}[r] = \phi(x_t, a_t)^{\mathsf{T}} \theta$
 - This is what we did in the linear MDP model! Helpful especially when K is large, since in that case there would be a lot of $\theta^{(k)}$ to fit

Both cases allow a version of linUCB by extension of the same ideas: fit coefficients via least squares and use Chebyshev-like uncertainty quantification to get UCB









More detail on the combined linear model

For $t = 0 \rightarrow T - 1$ 1. $\forall k$, define $A_t = \sum_{t=1}^{t-1} \phi(x_t, a_t) \phi(x_t,$ $\tau = 0$

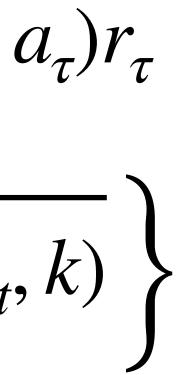
2. Observe x_t & choose $a_t = \arg \max_{t \in I_t} a_t$ 3. Observe reward $r_t \sim \nu^{(a_t)}(x_t)$

Comments:

- i.
- ii. Good for large *K*, but step 2's argmax may be hard

$$(x_{\tau}, a_{\tau})^{\top} + \lambda I \quad \text{and} \quad \hat{\theta}_{t} = A_{t}^{-1} \sum_{\tau=0}^{t-1} \phi(x_{\tau}, d_{\tau})$$
$$= A_{t}^{-1} \sum_{\tau=0}^{t-1} \phi(x_{\tau}, d_{\tau}) \sum_{\tau=0}^{t-1} \phi(x_{\tau}, d_{\tau})$$

There is only one A_t and $\hat{\theta}_t$ (not one per arm), so more info shared across k

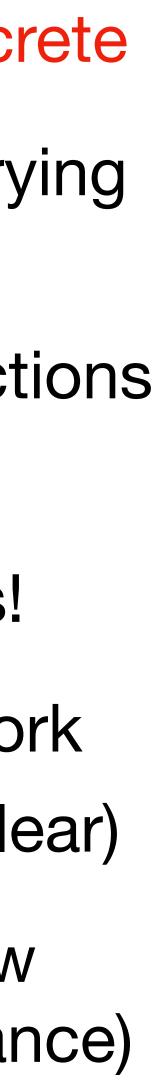




Continuous bandit action spaces

- This is because we to some extent treated each arm separately, necessitating trying each arm at least a fixed number of times before real learning could begin
- But now with the new combined formulation, there is sufficient sharing across actions that we can learn $\hat{\theta}_t$ and its UCB without sampling all arms
 - This means that in principle, we can now consider continuous action spaces!
 - This is the power of having a strong model for $\mathbb{E}_{r \sim \nu^{(a_t)}(x_t)}[r]$, and a neural network would serve a similar purpose in place of the combined linear model (UQ less clear)
- But in principle, there is no "free lunch", i.e., the hardness of the problem now transfers over to choosing a good model (a bad model will lead to bad performance)

In bandits / contextual bandits, we have always treated the action space as discrete







- Recap
- UCB-VI for linear MDPs
- Recall: Contextual Bandits
- LinUCB

Summary:

- Modeling in MDPs and bandits with large state/action spaces is critical
- and balance exploration/exploitation

Attendance: bit.ly/3RcTC9T



• When model is linear (in feature space), can still rigorously quantify uncertainty

Feedback: bit.ly/3RHtlxy



