Exploration: Contextual Bandits

Lucas Janson and Sham Kakade

CS/Stat 184: Introduction to Reinforcement Learning

Fall 2022

Today

- Recap
- LinUCB algorithm for contextual bandits

Recap: Bandits + confidence bounds

For $t = 0 \to T - 1$

1. Learner pulls arm $a_t \in \{1,\ldots,K\}$ (# based on historical information)
2. Learner observes an i.i.d reward $r_t \sim \nu_{a_t}$ of arm a_t

Note: there is no state s; rewards from a given arm are i.i.d. (data NOT i.i.d.!)

$$\mu^{(k)} = \mathbb{E}_{r\sim\nu_k}[r], \qquad N_t^{(k)} = \sum_{\tau=0}^{t-1} 1_{\{a_\tau=k\}}, \qquad \hat{\mu}_t^{(k)} = \frac{1}{N_t^{(k)}} \sum_{\tau=0}^{t-1} 1_{\{a_\tau=k\}} r_\tau$$

Uniform confidence bounds via Hoeffding + Union Bound

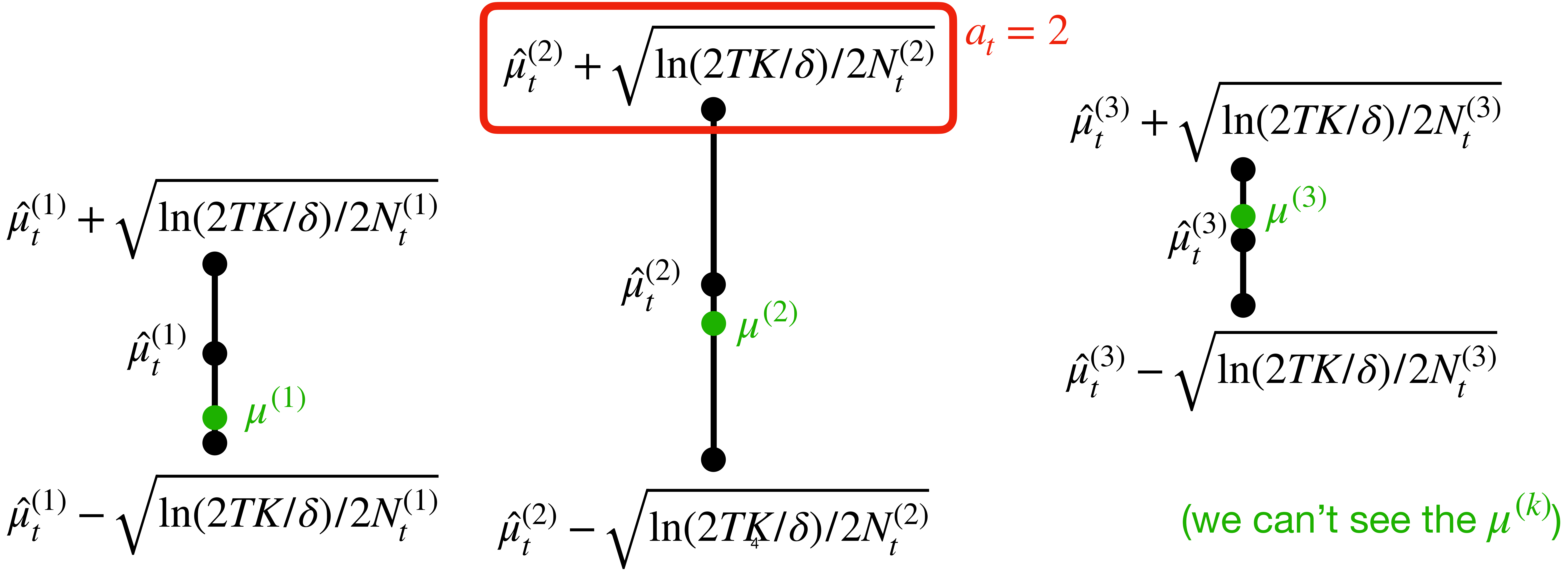
$$\mathbb{P}\left(\forall k \leq K, t < T, |\hat{\mu}_t^{(k)} - \mu^{(k)}| \leq \sqrt{\ln(2TK/\delta)/2N_t^{(k)}}\right) \geq 1 - \delta$$

Recap: Upper Confidence Bound (UCB) algorithm

For $t = 0, \ldots, T-1$:

Choose the arm with the highest upper confidence bound, i.e.,

$$a_t = \arg\max_{k \in \{1,\ldots,K\}} \hat{\mu}_t^{(k)} + \sqrt{\ln(2TK/\delta)/2N_t^{(k)}}$$



Recap: *optimism in the face of uncertainty*

Optimism in the face of uncertainty is an important principle in RL.
It basically says to give each arm the benefit of the doubt, and basically act as if that arm is as good as it could plausibly be in choosing an action

In UCB, this means constructing a CI (i.e., set of plausible values) for each $\mu^{(k)}$, and being greedy with respect to the upper bound of the CIs

Since each upper bound is $\hat{\mu}_t^{(k)} + \sqrt{\ln(2KT/\delta)/2N_t^{(k)}}$, this means when we select $a_t = k$, at least one of the two terms is large, i.e., either

1. $\sqrt{\ln(2KT/\delta)/2N_t^{(k)}}$ large, i.e., we haven't explored arm k much (exploration)
2. $\hat{\mu}_t^{(k)}$ large, i.e., based on what we've seen so far, arm k is the best (exploitation)

Note that the exploration here is *adaptive*, i.e., focused on most promising arms

Recap: Contextual bandit environment

For $t = 0 \to T - 1$

1. Learner sees context $x_t \sim \nu_x; x_t \in \mathbb{R}^d$ Independent of any previous data
2. Learner pulls arm $a_t = \pi_t(x_t) \in \{1, \ldots, K\}$ 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

Note that if the context distribution ν_x always returns the same value (e.g., 0), then the contextual bandit reduces to the original multi-armed bandit

Contextual bandit is exactly a MDP with horizon $H = 1$, where x_t is the (singular) state in each episode (so $\mu_0 = \nu_x$)

Recap: UCB in tabular contextual bandits

UCB algorithm also conceptually identical as long as $|\mathcal{X}|$ finite:

$$\pi_t(x_t) = \arg\max_k \hat{\mu}_t^{(k)}(x_t) + \sqrt{\ln(2TK|\mathcal{X}|/\delta)/2N_t^{(k)}(x_t)}$$

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

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

<u>Example</u>: 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!

Recap: Modeling in contextual bandits

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

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) $\Rightarrow x \in \{0,1\}^2$

$|\mathscr{X}| = 4$ $\Rightarrow$ 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 $\theta^{(k)} \in \mathbb{R}^2$

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

Choosing the best model, fitting it, and quantifying uncertainty are essentially problems of supervised learning (for another day)

Today

- ✓ Recap
- LinUCB algorithm for contextual bandits

Linear model fitting

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

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

Least squares estimator: $\hat{\theta}_t^{(k)} = \arg\min_{\theta \in \mathbb{R}^d} \sum_{\tau=0}^{t-1} (r_\tau - x_\tau^\top \theta)^2 1_{\{a_\tau = k\}}$

Minimize squared error over time points when arm k selected

$$\text{Claim: } \hat{\theta}_t^{(k)} = \left(\sum_{\tau=0}^{t-1} x_\tau x_\tau^\top 1_{\{a_\tau = k\}} \right)^{-1} \sum_{\tau=0}^{t-1} x_\tau r_\tau 1_{\{a_\tau = k\}}$$

proof:

Linear model fitting (cont'd)

Recall: $\hat{\theta}_t^{(k)} = \left(\sum_{\tau=0}^{t-1} x_\tau x_\tau^\top 1_{\{a_\tau = k\}} \right)^{-1} \sum_{\tau=0}^{t-1} x_\tau r_\tau 1_{\{a_\tau = k\}}$

Let $A_t^{(k)} = \sum_{\tau=0}^{t-1} x_\tau x_\tau^\top 1_{\{a_\tau = k\}}$ and $b_t^{(k)} = \sum_{\tau=0}^{t-1} x_\tau r_\tau 1_{\{a_\tau = k\}}$

Then $\hat{\theta}_t^{(k)} = \left(A_t^{(k)} \right)^{-1} b_t^{(k)}$

$A_t^{(k)}$ like empirical covariance matrix of the contexts when arm k was chosen

$b_t^{(k)}$ 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 confidence bounds 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 → better estimation

But not using sample mean as estimator, so need something other than Hoeffding

Chebyshev's inequality: for a mean-zero random variable Y,

$$|Y| \leq \beta\sqrt{\mathbb{E}[Y^2]} \quad \text{with probability } \geq 1 - 1/\beta^2$$

Uncertainty quantification (cont'd)

Want confidence bounds on our estimated mean rewards for each arm: $x_t^\top \hat{\theta}_t^{(k)}$

Strategy: apply Chebyshev's inequality to $x_t^\top \hat{\theta}_t^{(k)} - x_t^\top \theta^{(k)}$

Need: $\mathbb{E}[x_t^\top \hat{\theta}_t^{(k)} - x_t^\top \theta^{(k)}]$ (make sure it's zero) and $\mathbb{E}\left[(x_t^\top \hat{\theta}_t^{(k)} - x_t^\top \theta^{(k)})^2\right]$

Let $w_t = r_t - \mathbb{E}_{r \sim \nu^{(k)}(x_t)}[r] = r_t - x_t^\top \theta^{(k)}$, and we derive a useful expression for $\hat{\theta}_t^{(k)}$:

Uncertainty quantification (cont'd)

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

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

$$\mathbb{E}_{w_\tau}[x_t^\top \hat{\theta}_t^{(k)} - x_t^\top \theta^{(k)}] = \mathbb{E}_{w_\tau}[x_t^\top (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_\tau 1_{\{a_\tau = k\}} w_\tau] = x_t^\top (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_\tau 1_{\{a_\tau = k\}} \mathbb{E}_{w_\tau}[w_\tau] = 0$$

$$\mathbb{E}_{w_\tau}[(x_t^\top \hat{\theta}_t^{(k)} - x_t^\top \theta^{(k)})^2] = \mathbb{E}_{w_\tau}\left[\left(x_t^\top (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_\tau 1_{\{a_\tau = k\}} w_\tau\right)^2\right]$$

$$= x_t^\top (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} \sum_{\tau'=0}^{t-1} x_\tau x_{\tau'}^\top 1_{\{a_\tau = k\}} 1_{\{a_{\tau'} = k\}} \mathbb{E}_{w_\tau}\left[w_\tau w_{\tau'}\right] (A_t^{(k)})^{-1} x_t$$

$$= x_t^\top (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_\tau x_\tau^\top 1_{\{a_\tau = k\}} \mathbb{E}_{w_\tau}[w_\tau^2] (A_t^{(k)})^{-1} x_t \le x_t^\top (A_t^{(k)})^{-1} A_t^{(k)} (A_t^{(k)})^{-1} x_t = x_t^\top (A_t^{(k)})^{-1} x_t$$

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 $\geq 1 - 1/\beta^2$

Intuition:

UCB term 1: $x_t^\top \hat{\theta}^{(k)}$ large when context and coefficient estimate aligned

UCB term 2: $x_t^\top (A_t^{(k)})^{-1} x_t = \frac{1}{N_t^{(k)}} x_t^\top (\Sigma_t^{(k)})^{-1} x_t$, where

$\Sigma_t^{(k)} = \frac{1}{N_t^{(k)}} A_t^{(k)} = \frac{1}{N_t^{(k)}} \sum_{\tau=0}^{t-1} x_\tau x_\tau^\top 1_{\{a_\tau = k\}}$ 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!

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…

Issue 2: $A_t^{(k)}$ has to be invertible

Before the dth time that arm k gets pulled, $\hat{\theta}_t^{(k)}$ undefined

Solution (to both issues): regularize

Replace $A_t^{(k)} \leftarrow A_t^{(k)} + \lambda I$ for some $\lambda > 0$

Makes $A_t^{(k)}$ 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

For $t = 0 \to T-1$

1. $\forall\, k$, define $A_t^{(k)} = \sum_{\tau=0}^{t-1} x_\tau x_\tau^\top 1_{\{a_\tau = k\}} + \lambda I$ and $\hat{\theta}_t^{(k)} = (A_t^{(k)})^{-1} \sum_{\tau=0}^{t-1} x_\tau r_\tau 1_{\{a_\tau = k\}}$
2. Observe context x_t and choose $a_t = \arg\max_k \left\{ x_t^\top \hat{\theta}_t^{(k)} + c_t \sqrt{x_t^\top (A_t^{(k)})^{-1} x_t} \right\}$
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.,

$$\mathbb{E}_{r \sim \nu^{a_t}(x_t)}[r] = \phi(x_t, a_t)^\top \theta$$

   This is what problem 3 of HW 1 (which we cut) was about; it's helpful especially when K is large, since in that case there are 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 $t = 0 \to T-1$

1. $\forall k$, define $A_t = \sum_{\tau=0}^{t-1} \phi(x_\tau, a_\tau)\phi(x_\tau, a_\tau)^\top + \lambda I$ and $\hat{\theta}_t = A_t^{-1} \sum_{\tau=0}^{t-1} \phi(x_\tau, a_\tau) r_\tau$
2. Observe x_t & choose $a_t = \arg\max_k \left\{ \phi(x_t, k)^\top \hat{\theta}_t + c_t \sqrt{\phi(x_t, k)^\top A_t^{-1} \phi(x_t, k)} \right\}$
3. Observe reward $r_t \sim \nu^{(a_t)}(x_t)$

Comments:

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

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

iii. The other formulation, with separate $A_t^{(k)}$ and $\hat{\theta}_t^{(k)}$, is called disjointed

Continuous bandit action spaces

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

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)

Today

- ✓ Recap
- ✓ LinUCB algorithm for contextual bandits

Today's summary:

LinUCB algorithm for contextual bandits

- Uses UCB idea, but requires modeling reward distribution
- Uses Chebyshev's inequality for uncertainty quantification

Next time:

- UCB-VI: apply UCB idea to full (tabular) RL (essentially a contextual bandit with continuous and highly structured action space)

1-minute feedback form: https://bit.ly/3RHtlxy