CSE 547/Stat 548: Machine Learning for Big Data **Lecture**

Thompson Sampling and Linear Bandits

Instructor: Sham Kakade

1 Review

The basic paradigm is as follows:

- K Independent Arms: $a \in \{1, \dots K\}$
- Each arm a returns a random reward R_a if pulled.
  (simpler case) assume R_a is not time varying.
- Game:
  - You chose arm a_t at time t.
  - You then observe:
    $$X_t = R_{a_t}$$
    where R_{a_t} is sampled from the underlying distribution of that arm.

Critically, the distribution over R_a is not known.

2 Thompson Sampling a.k.a. Posterior Sampling

Our history of information is:

$$\text{History}_{<t} = (a_1, X_1, a_2, X_2, \dots a_{t-1}, X_{t-1})$$

One practical question is how to obtain good confidence intervals? Here, often Bayesian methods work quite well. If we were Bayesian, we would actually have a posterior distribution of the form:

$$\Pr(\mu_a | \text{History}_{<t})$$

which specifies our belief about the what μ_a could be given our history of information.

If we were truly Bayes optimal, then we use our posterior beliefs to design an algorithm which actives the minimal Bayes regret (such as in Gittins index algorithm).

Instead, Thompson sampling is a simple way to do something reasonable, which is near to optimal (in a minimax sense) in many cases, much like UCB is minimax optimal. The algorithm is as follows:

For each time t,

1. Sample from each posterior:
   $$\nu_a \sim \Pr(\mu_a | \text{History}_{<t})$$

2. take action

$$a_t = \arg\max_a \nu_a$$

3. update our posteriors and go back to 1.

Regret of the Posterior Sampling: In a multi-armed bandit setting (just like for UCB) and under some restriction on our prior, the total expected regret of Thompson sampling is identical to that of the UCB:

$$\mu_* T - \mathbb{E}\left[\sum_{t=1}^{T} X_t\right] \leq c\sqrt{KT \log T}$$

for an appropriately chosen universal constant c. See the related readings for this discussion.

3 Linear Bandits

In practice, our space of actions might be very large. The most common way to address this is attempt to embed this space so that there is a linear structure in the reward function.

3.1 The Setting

One can view the linear bandits model as an additive effects model (a regression model), where at each round we take a decision $x \in \mathcal{D} \subset \mathbb{R}^d$ and our payout is linear in this decision.

Examples include:

- x is path on a graph.
- x is a feature vector of properties of an ad
- x is which drugs are being prescribed.

Upon taking action x, we observe reward r, with expectation:

$$\mathbb{E}[r|x] = \mu^\top x$$

Here, we only have d unknown parameters (and "effectively" 2^d actions). As before, we desire an algorithm $\mathcal{A}$ (mapping histories to decisions), which has low regret.

$$T\mu^\top x_* - \sum_{t=1}^{T} \mathbb{E}[\mu^\top x_t|\mathcal{A}] \leq ?$$

(where x_* is the best decision)

3.2 The Algorithm: LinUCB

Again, let's think of optimism in the face of uncertainty! We have observed some $r_1, \ldots r_{t-1}$, and have taken $x_1, \ldots x_{t-1}$. Questions:

- what is an estimate of the reward of $\mathbb{E}[r|x]$ and what is our uncertainty?
- what is an estimate of μ and what is our uncertainty?

We can address these issues using our understanding of regression: Define:

$$A_t := \sum_{\tau<t} x_\tau x_\tau^\top + \lambda I,\ b_t := \sum_{\tau<t} x_\tau r_\tau$$

Our estimate of μ is:

$$\hat{\mu}_t = A_t^{-1} b_t$$

and a valid confidence of our estimate:

$$\|\mu - \hat{\mu}_t\|_{A_t}^2 \leq \mathcal{O}(d \log t)$$

(which will hold with probability greater than $1 - \text{poly}(1/t)$).

The algorithm: Define:

$$B_t := \{\nu | \|\nu - \hat{\mu}_t\|_{A_t}^2 \leq \mathcal{O} d \log t\}$$

- At each time t, take action:

  $$x_t = \arg\max_{x \in \mathcal{D}} \max_{\nu \in B_t} \nu^\top x$$

  then update A_t, B_t, b_t, and $\hat{\mu}_t$.
- Equivalently, take action:

  $$x_t = \arg\max_{x \in \mathcal{D}}\ \hat{\mu}_t^\top x + (d \log t)\sqrt{x A_t^{-1} x}$$

3.3 Regret

Theorem 3.1. *The expected regret bound of LinUCB is bounded as:*

$$T\mu^\top x_* - \sum_{t=1}^{T} \mathbb{E}[\mu^\top x_t] \leq O^*(d\sqrt{T})$$

(this is the best possible, up to log factors).

A few points:

- compare this to $O(\sqrt{KT})$ for the k-arm case
- This bound is *independent of number of actions.*
- k-arm case is a special case.
- One can also do Thompson sampling as variant of LinUCB, which is a reasonable algorithm in practice.