IEOR 8100-001: Learning and Optimization for Sequential Decision Making 02/17/16

Lecture 9: Linear Bandits (Part II)

Instructor: Shipra Agrawal Scribed by: Yuanjun Gao

1 UCB Algorithm for Linear Bandit Setting

1.1 Setting and Algorithm

Under the linear bandit setting, at time t, we are given a set of accessible bandits $A_t \subseteq A \subset \mathbb{R}^d$. We pick $x_t \in A_t$ and observe r_t. We have $E[r_t|x_t] = w'x_t$, where $w \in \mathbb{R}^d$ is a unknown fixed vector. Define regret as

$$R(T) = \sum_{t=1}^{T}(\max_{x_t^* \in A_t} w'x_t^*) - \sum_{t=1}^{T} w'x_t$$

The UCB algorithm for linear bandit problem proceeds as follows. At each time t we obtain a (regularized) least square estimator for w using all past observations

$$\hat{w}_t = \arg\min_z \sum_{s=1}^{t}(r_s - z'x_s)^2 + \|z\|^2 = M_t^{-1} y_t$$

where

$$M_t = I + \sum_{s=1}^{t} x_s x_s', \quad y_t = \sum_{s=1}^{t} x_s r_s$$

There exist an elliptical confidence region for the w, as described in the following theorem

Theorem 1. *([2], Theorem 2) Assuming $\|w\| \leq \sqrt{d}$ and $\|x_t\| \leq \sqrt{d}$, with probably $1-\delta$, we have $w \in C_t$, where*

$$C_t = \left\{ z : \|z - \hat{w}_t\|_{M_t} \leq 2\sqrt{d \log \frac{Td}{\delta}} \right\}$$

For any $x \in A$, we define $\text{UCB}_{x,t} = \max_{z \in C_t} z'x$ if $w \in C_t$ (which holds with high probability). At each time, the UCB algorithm then simply picks the bandit with the highest UCB given all previous observation.

$$x_t = \arg\max_{x \in A} \text{UCB}_{x,t-1} = \arg\max_{x \in A, z \in C_{t-1}} x'z$$

For the regular multi-armed bandit setting, we provide a sketch for a simple analysis for the order of the regret.

$$
\begin{aligned}
R(T) =& \sum_{t=1}^{T} (\mu_t^* - \mu_{I_t}) && (1)\\
\leq& \sum_{t=1}^{T} \mathrm{UCB}_{I_t^*, t-1} - \mu_{I_t} && (2)\\
\leq& \sum_{t=1}^{T} \mathrm{UCB}_{I_t, t-1} - \mu_{I_t} && (3)\\
=& \sum_{i=1}^{N} \sum_{t: I_t = i} \sqrt{\frac{\log T}{n_{i,t-1}}} && (4)\\
=& \sum_{i=1}^{N} \sum_{k=1}^{N_{i,T}} \sqrt{\frac{\log T}{k}} && (5)\\
=& \sqrt{\log T} \sum_{i} \sqrt{n_{i,T}} && (6)\\
\leq& \sqrt{\log T} \sqrt{NT} && (7)
\end{aligned}
$$

(1) comes from definition. (2) hold with high probability since $\mathrm{UCB}_{I_t^*, t-1} > \mu_t^*$ with high probability. (3) holds by definition of the UCB algorithm (i.e. we pick the bandit with the highest UCB). (4) holds because $UCB - \mu$ are bounded by $\sqrt{\frac{\log T}{n_{I_t, t-1}}}$. (5) is a rearrangement of (4) by noting that each time arm i has the highest UCB, it will be pulled one more time, so $n_{i,t}$ increases by 1. (6) uses $\sum_{i=1}^{n} \frac{1}{\sqrt{i}} = O(\sqrt{n})$. (7) holds because $n_{i,T} = \frac{T}{n}$ gives the worst case.

We adapt this idea to the linear bandit case by noting

$$
\begin{aligned}
R(T) \leq& \sum_{t=1}^{T} w' x_t^* - w' x_t \\
=& \sum_{t=1}^{T} \mathrm{UCB}_{x_t^*, t-1} - w' x_t \\
\leq& \sum_{t=1}^{T} \mathrm{UCB}_{x_t, t-1} - w^T x_t && (8)
\end{aligned}
$$

Here we have $\mathrm{UCB}_{x_t, t} = z_{t-1}' x_t$ for some $z_{t-1} \in C_t$, where $\|z_{t-1} - w\|_{M_t} \leq 2\sqrt{d \log(dT/\delta)}$ with probability $1 - \delta$. We proceed by

$$
\begin{aligned}
(8) =& \sum_{t=1}^{T} z_{t-1}' x_t - w' x_t \\
\leq& \sum_{t=1}^{T} \|z_{t-1} - w\|_{M_{t-1}} \|x_t\|_{M_{t-1}^{-1}} && (9)\\
\leq& 2\sqrt{d \log(dT/\delta)} \sum_{t=1}^{T} \|x_t\|_{M_{t-1}^{-1}} && (10)
\end{aligned}
$$

Here (9) comes from Cauchy-Schwarz inequality ($|x'w| \leq \|x\|_{M^{-1}} \|w\|_M$). (10) is because, as mentioned above, $\|z_{t-1} - w\|_{M_t} \leq 2\sqrt{d \log(Td/\delta)}$ holds with probability $1 - \delta$.

Now we want to get something similar to (6) to bound the summation $\sum_{t=1}^{T} \|x_t\|_{M_{t-1}^{-1}} = \sum_{t=1}^{T} \sqrt{x_t' M_{t-1}^{-1} x_t}$. The tricky thing is that although M_t keeps increasing, there are many directions in $M_t \in \mathbb{R}^{d\times d}$, so even for large t, if x_t is in the direction of a eigenvector of M_{t-1} with a small eigenvalue, $\|x_t\|_{M_{t-1}^{-1}}$ can still be large. Fortunately, we have the following lemma

Lemma 2. *(Lemma 11 of [3], or, Lemma 2 of [4]) Denote $\lambda_{j,t-1}$ as the j^{th} largest eigenvalue of M_{t-1}, then eigenvalues of M_t can be arranged so that $\lambda_{j,t} \geq \lambda_{j,t-1}$, and we have*

$$\|x_t\|^2_{M_{t-1}^{-1}} \leq 10 \sum_{j=1}^{d} \frac{\lambda_{j,t} - \lambda_{j,t-1}}{\lambda_{j,t-1}}$$

Intuitively, this lemma shows that if x_t is in the direction of a eigenvector of M_{t-1} with a small eigenvalue, then, it will sufficiently increase that eigenvalue, which would benefit that direction in the next time step. Therefore, in any direction we will get decreasing terms in the summation. More precisely, we have

$$(10) \leq 2\sqrt{d\log(Td/\delta)} \sum_{t=1}^{T} \sqrt{\sum_j \left(\frac{\lambda_{j,t}}{\lambda_{j,t-1}} - 1\right)} \tag{11}$$

The remaining analysis involves considering the worst possible value (to maximize above expression) of $\lambda_{j,t}, j, t$ under the constraint $\sum_j \Pi_{t=1}^{T} \frac{\lambda_{j,t}}{\lambda_{j,t-1}} = \sum_j \lambda_{j,T} \leq T$, and $\frac{\lambda_{j,t}}{\lambda_{j,t-1}} \geq 1$. It can be shown (refer to [4]: Lemma 3 in Section 5) that at maximizer $h_{tj} := \frac{\lambda_{j,t}}{\lambda_{j,t-1}}$ are equal for all t, j and $\sum_{t=1}^{T} \sqrt{\sum_j \left(\frac{\lambda_{j,t}}{\lambda_{j,t-1}} - 1\right)} \leq O(\sqrt{dT\ln(T)})$, so that assuming $d \leq T$

$$(10) \leq O(\sqrt{d\log(Td/\delta)}\sqrt{dT\ln(T)}) = O(d\sqrt{T\log^2(T/\delta)}) \tag{12}$$

This proves that regret of this UCB algorithm for linear bandits is

$$R(T) \leq O(d\sqrt{T\log^2(T/\delta)})$$

with probability $1 - \delta$.

2 Adversarial case

2.1 Definition

Here we want to pick $x_t \in A$ each time to maximize the reward (here A is not time-varying, and we assume it to behave well. For example, we assume that it is a convex set), and we assume that at each time our expected reward is $x_t' w_t$, where the weight changes across time (and have no pattern)

In this case we compare our strategy with the best strategy that keeps pulling one single arm. So we define regret as

$$R(T) = \left(\max_{x\in A} \sum_t x' w_t\right) - \sum_{t=1}^{T} x_t' w_t$$

2.2 Full information setting

Under the full information setting, we observe w_t after picking x_t. We can use the simple idea of online linear optimization by gradient ascent. Notice that the reward is $r_t(x) = x' w_t$, so the gradient is simply $\frac{dr_t(x)}{dx} = w_t$.

Therefore we want our x_t go in the direction of w_t a little. Therefore we update our choice by

$$x_t = \Pi_A(x_{t-1} + \eta w_{t-1})$$

where Π_A is the projection operator and η is a constant step-size. We have

Algorithm 1 Gradient Ascent Algorithm for Full Information Linear Bandit under Adversarial Case

```
Input $\eta > 0$
for t = 1,2,... do
    $x_t = \Pi_A(x_{t-1} + \eta w_{t-1})$
    Play arm $x_t$, observe reward $r_t$ and $w_t$
end for
```

Theorem 3. *Under the full information setting, assuming* $\|w_t\| \leq \sqrt{d}$*,* $\forall x \in A, \|x\| \leq \sqrt{d}$*, then using the gradient ascent algorithm with* $\eta = \frac{1}{\sqrt{T}}$*, we have*

$$R(T) \leq d\sqrt{T}$$

More generally, suppose $\|w_t\| \leq D$*,* $\forall x \in A, \|x\| \leq G$*, then we have*

$$R(T) \leq DG\sqrt{T}$$

Note that in the adversarial N-armed bandit setting, we have a $\sqrt{T \log N}$ bound.

2.3 Bandit setting

Suppose that instead of observing w, we only get to observe $r_t = w_t' x_t$ after picking x_t, we can adapt the gradient ascent algorithm for the full information setting by using an unbiased estimator for w_{t-1}. Here, instead of pulling x_t, we perturb it a little by a random walk. Specifically, we generate a random vector $u \in \mathbb{R}^d$, where each element u_i is generated independently and equals 1 or -1 with probability $1/2$. Then we pull arm $x_t + \delta u$ to get the reward. Interestingly, this random perturbation gives us an unbiased estimator for w

Claim 4. *The* $\hat{w}_t$ *defined below is an unbiased estimator of* w_t

$$\hat{w}_t = w_t' \frac{(x_t + \delta u)u}{\delta}$$

Proof.

$$\begin{aligned} E[\hat{w}_t] =& E[\frac{w_t' x_t u}{\delta}] + E[uu' w_t] \\ =& 0 + E[I_d w_t] \\ =& 0 + w_t \end{aligned}$$

Here we use the fact that $E(u) = 0$ (since it is a random walk) and $E[uu'] = I_d$ (since u_i are independent and $u_i^2 = 1$ with probability 1) □

So in sum, in the bandit setting, we use the following update rule

$$x_t = \Pi_A(x_{t-1} + \eta \hat{w}_{t-1})$$

But each time we actually pull $x_t + \delta u_t$ for a random vector u_t.

By using an unbiased estimator instead of the true w_t, we sacrifice in the following two ways.

Algorithm 2 Gradient Ascent Algorithm for Linear Bandit Setting under Adversarial Case

```
Input $\eta > 0$, $\delta > 0$
for t = 1,2,... do
    $x_t = \Pi_A(x_{t-1} + \eta\hat{w}_{t-1})$
    Play $x_t + \delta u_t$, where $u_t$ is a random vector
    Observe $r_t$
    Define $\hat{w}_t = w_t' \frac{(x_t+\delta u)u}{\delta}$
end for
```

First the $\hat{w}_t$ can be big, so we have to increase the D in theorem 3. Actually now we have $D = \frac{\sqrt{d}}{\delta} \geq \|\hat{w}_t\|$, which increase the bound by $\frac{1}{\delta}$ fold. Giving us $\frac{d\sqrt{T}}{\delta}$

Secondly, instead of pulling x_t, we have the random perturbation δu, which adds an extra regret

$$\sum_t \delta u_t' w_t \leq \delta d T$$

This is because we have

$$w_t'(x_t + \delta u) \leq w_t' x_t - \delta|w_t' u|$$

but $|w_t' u| \leq d$ because $\|u\| = \sqrt{d}$ and we assume $\|w_t\| \leq \sqrt{d}$.

Combining the above two point together, we get a bound of the form

$$\frac{d\sqrt{T}}{\delta} + \delta d T$$

By setting $\delta = \frac{1}{T^{1/4}}$, we get a lower bound of

$$R(T) = O(dT^{3/4})$$

The optimal lower bound has been proved to be $\Omega(d\sqrt{T})$, which cannot be achieved by the algorithm stated above. The algorithm with the optimal rate involves a much more complicated algorithm.

[5] provides analysis of online gradient ascent algorithm for full information setting, [6] extends it to bandit setting. [7] provides efficient algorithm which achieves a regret upper bound with optimal dependence of $\sqrt{T}$ on time horizon T.

References

[1] Stochastic Linear Optimization under Bandit Feedback, Varsha Dani, Thomas P. Hayes, Sham M. Kakade, COLT 2008.

[2] Improved Algorithms for Linear Stochastic Bandits, Yasin Abbasi-yadkori, Dvid Pl, Csaba Szepesvri, NIPS 2011.

[3] Using Confidence Bounds for Exploitation-Exploration Trade-offs, Peter Auer. JMLR 3(Nov):397-422, 2002.

[4] Contextual Bandits with Linear Payoff Functions. Wei Chu. Lihong Li. Lev Reyzin. Robert E. Schapire. AISTATS 2011.

[5] Online Convex Programming and Generalized Infinitesimal Gradient Ascent, Martin Zinkevich. ICML 2003.

[6] Online convex optimization in the bandit setting: gradient descent without a gradient, Abraham D. Flaxman, Adam Tauman Kalai, H. Brendan McMahan. SODA 2005.

[7] Competing in the Dark: An Efficient Algorithm for Bandit Linear Optimization by Jacob Abernethy , Elad Hazan , Alexander Rakhlin, COLT 2008.