STATS 710 – Seq. Dec. Making with mHealth Applications Nov 17, 2016

Lecture 19: PAC(Probably Approximately Correct) learning

Instructors: Susan Murphy and Ambuj Tewari *Scribe: April Cho*

1 PAC (Probably Approximately Correct) learning

- PAC learning basic idea: At the end of the ”learning” process, we want the “solution” that should, with high probability, be approximately correct.
- In multi-armed bandits, $\mathcal{A} = \{0, 1, \cdots k-1\}, \mu_a = \mathbb{E}[R^a]$, and $R^a \in [0,1]$. For each a, we have access to an iid stream $R_1^a, R_2^a, \cdots$. The optimal arm is $a^\star = \text{argmax}_{a \in \mathcal{A}} \mu_a$, and $\mu^\star = \max_{a \in \mathcal{A}} \mu_a$.

Algorithm 1 is a general template of the PAC algorithm for MAB [EDMM02]. We need to specify the termination condition and sampling method for it to be a proper algorithm.

Algorithm 1 PAC learning for Multi-Armed Bandits

```
At time t :
if TERMINATION CONDITION met then
    Terminate and output an arm $A'$
else
    Sample an arm $A_t$
    Receive $R_t = R_t^{A_t}$
end if
```

1.1 Definitions

Here are some definitions we need to know for the following theorems and proof.

- Arm/action a is *ϵ-optimal* if $\mu_a \geq \mu^\star - \epsilon$
- A learning algorithm is an *(ϵ, δ)-PAC algorithm* for MAB if it outputs an ϵ-optimal arm with probability $\geq 1 - \delta$
- *Sample Complexity* of a learning algorithm for MAB = Worst case(over all possible inputs) time steps needed until termination

1.2 NAIVE algorithm

Sampling each action same amount of time is the most naive way of sampling actions.

Algorithm 2 NAIVE algorithm

```
For each $a \in \mathcal{A}$, sample $a$ a total of $l$ times
Let $\hat{\mu}_a$ be the sample means corresponding to the K actions
Terminate and output $A' = \text{argmax}_{a \in \mathcal{A}} \hat{\mu}_a$
```

Sample complexity of NAIVE algorithm is $l \cdot K$ (fixed).

Theorem 1. *Naive algorithm with $l = \frac{4}{\epsilon^2}\log(\frac{2K}{\delta})$ is (ϵ, δ)-PAC. Therefore for this choice of l, its sample complexity is $O(\frac{K}{\epsilon^2}\log(\frac{K}{\delta}))$.*

NOTE : There is a lower bound on sample complexity of $\Omega(\frac{K}{\epsilon^2}\log(\frac{1}{\delta}))$.

Proof. Let a be some action that is not ϵ-optimal : $\mu_a < \mu^\star - \epsilon$

$$\begin{aligned}
\mathbb{P}(A' = a) &\leq \mathbb{P}(\hat{\mu}_a \geq \hat{\mu}_{a^\star}) \\
&= \mathbb{P}(\hat{\mu}_a \geq \mu_a + \frac{\epsilon}{2} \text{ or } \hat{\mu}_{a^\star} \leq \mu^\star - \frac{\epsilon}{2}) \\
&\leq \mathbb{P}(\hat{\mu}_a \geq \mu_a + \frac{\epsilon}{2}) + \mathbb{P}(\hat{\mu}_{a^\star} \leq \mu^\star - \frac{\epsilon}{2}) \\
&\leq e^{-(\frac{\epsilon}{2})^2 l} + e^{-(\frac{\epsilon}{2})^2 l} \\
&= \frac{\delta}{K} \qquad \text{when } l = \frac{4}{\epsilon^2}\log(\frac{2K}{\delta})
\end{aligned}$$

Since we have K actions, $\mathbb{P}$(NAIVE outputs a non ϵ-optimal arm) $\leq K \cdot \delta/K = \delta$. Hence with $l = \frac{4}{\epsilon^2}\log(\frac{2K}{\delta})$, NAIVE algorithm is ($\epsilon$, δ)-PAC. □

1.3 MEDIAN ELIMATION(ϵ,δ)

ϵ and δ are the input variables for this algorithm. S_l is a collection of competing arms at l^{th} phase.

Algorithm 3 Median Elimination(ϵ,δ) algorithm

```
Set $S_1 = \mathcal{A}$
$\epsilon_1 = \frac{\epsilon}{4}$, $\delta_1 = \frac{\delta}{2}$, and $l = 1$.
while $|S_l| > 1$ do
    Sample every arm $a \in \mathcal{S}_l$ for $\frac{1}{(\epsilon_l/2)^2}\log(\frac{3}{\delta_l})$ times
    Let $\hat{\mu}_a^l$ denote the sample mean using samples in the $l^{th}$ phase
    Find the median of the $|S_l|$ quantities $\mu_a^l =: m_l$
    $S_{l+1} = S_l - \{a : \hat{\mu}_a^l < m_l\}$
    $l = l + 1$; $\epsilon_{l+1} = \frac{3}{4}\epsilon_l$; $\delta_{l+1} = \frac{\delta_l}{2}$
end while
When out of the loop, output the only action remaining in $S_l$
```

Theorem 2. *Median Elimination(ϵ, δ) is (ϵ, δ)-PAC. Furthermore, its sample complexity is $O(\frac{K}{\epsilon^2}\log(\frac{1}{\delta}))$*

A key lemma is that, with high probability, the drop in the optimality of the best arm within the remaining set of arms is not severe in going from one phase to the next.

Lemma 3. *We have $\mathbb{P}(\max_{a \in S_l}\ \mu_a \leq \max_{a \in S_{l+1}}\ \mu_a + \epsilon_l) \geq 1 - \delta_l$*

We will prove this lemma next time.

Proof of Theorem 2(assuming Lemma 3 is true).

$$\begin{aligned}\text{Overall suboptimality of the action in the final set } S_l &= \epsilon_1 + \epsilon_2 + \epsilon_3 + \cdots + \epsilon_{\log_2(K)} \\ &\leq \sum_{i=1}^{\infty} \epsilon_i = O(\epsilon)\end{aligned}$$

Probability that inequality in Lemma 3 fails to hold in some episodes

$$\begin{aligned}&\leq \sum_{l=1}^{\log_2 K} \delta_l \\ &\leq \sum_{l=1}^{\infty} \delta_l = \delta\end{aligned}$$

$$\begin{aligned}\text{Sample Complexity} &= \sum_{l=1}^{\log_2 K} \frac{n_l \log(\frac{3}{\delta_l})}{(\epsilon_l/2)^2} \qquad\qquad \text{where } n_l = |S_l| = \frac{K}{2^{l+1}} \\ &= 4 \sum_{l=1}^{\log_2 K} \frac{(\frac{K}{2^{l-1}}) \log(\frac{2^l 3}{\delta})}{((\frac{3}{4})^{l-1} \frac{\epsilon}{4})^2} \\ &= 64 \sum_{l=1}^{\log_2 K} K(\frac{8}{9})^{l-1}(\frac{\log(1/\delta)}{\epsilon^2} + \frac{\log 3}{\epsilon^2} + l\frac{\log 2}{\epsilon^2}) \\ &\leq 64\frac{K \log(1/\delta)}{\epsilon^2} \sum_{l=1}^{\infty} (\frac{8}{9})^{l-1}(lC' + C) \\ &\leq O(\frac{K \log(1/\delta)}{\epsilon^2}).\end{aligned}$$

□

References

[EDMM02] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, pages 255–270. Springer, 2002.