Bandit algorithms in revenue optimization

(The value of knowing a demand curve – Kleinberg & Leighton)

- We now see how bandit algorithms (and in particular UCB) can be used to perform revenue optimization without knowing prices

- **Model** -
  - N buyers arrive sequentially
  - Seller makes a ‘posted price’ take-it-or-leave-it offer to each buyer.
  - Each buyer has i.i.d. value $V_t \sim F$, $E[0,1]$ a.s.
    - If posted price = $P_t$, then buyer purchases iff $P_t \leq V_t$
  - $F$ is unknown

- If $F$ is known, can use ‘monopolist price’
  \[
  P^* = \arg \max_P \left[ P \left[ 1 - F(p) \right] \right]
  \]
  - Assume $F$ is regular $\Rightarrow$ unique $P^*$, $R(p)$ quasi-concave
  - If instead we knew $V_1, V_2, \ldots, V_n$, can choose $P_{opt}$ to maximize revenue
* Thus - Assuming $R(p) = p \mathbb{F}(p)$ has unique global maximum $p^*$, and $R^*(p^*)$ exists and is strictly negative, then there is an online pricing strategy which achieves an expected regret of $O(\sqrt{n \ln n})$

Notes

* Let $\overline{R}_n^* = \mathbb{E}\left[\sum_{t=1}^{n} p^* 1\{\text{v}_t > p^*\}\right] = nR(p^*)$
  $\overline{R}_{\text{opt}} = \max_{p} \left[\sum_{t=1}^{n} p 1\{\text{v}_t > p^*\}\right]$ (Oracle bound)

  $\overline{R}_n = \mathbb{E}\left[\sum_{t=1}^{n} p_t 1\{\text{v}_t > p_t\}\right]$, where $p_t = \text{Policy } \Pi_t$

We will actually get that

$\overline{R}_n^* - \overline{R}_n^\Pi = O(\sqrt{n \ln n})$

and

$\overline{R}_{\text{opt}} - \overline{R}_n^* = O(\sqrt{n \ln n})$

Thus, we have small regret w.r.t. an oracle bound - this is stronger than competing with $p^*$

* Why regret? It was known that there are randomized pricing algos s.t. $\overline{R}_n^\Pi \geq \frac{1}{1+\varepsilon}$ for any $\varepsilon > 0$. Regret capture the lower order dependence on $n$

  This was the first regret bound for an ‘infinite’ arm setting
Proof - Main Idea - Choose appropriate discretization of [0,1]

- In particular, given K, we consider the price \( \{1/k, 2/k, \ldots, K/k\} \) as 'arms' (we later choose \( K = (n/\log n)^4 \))

- Now we can use UCB.

- For \( p_i = i/k \), the payoff is \( x_i = \begin{cases} i/k & \text{if } v_i > i/k \\ 0 & \text{otherwise} \end{cases} \)

\[ \Rightarrow \mu_i = E[x_i] = \frac{i}{k} \cdot F\left(\frac{i}{k}\right) \]

- Let \( (i^*, p^*) = \text{best arm } i/k \)

\[ \Delta_i = \left( p^* - p_i \right) \]

- We want to show we get close to \( np^* \), and also that \( np^* \) is close to \( np^* \cdot F(p^*) \).

Lemma: \( \exists \) constants \( C_1, C_2 \) s.t. \( C_1(p - p^*)^2 \leq R(p^*) \leq C_2(p - p^*)^2 \) for all \( p \in [0,1] \)

Proof - For \( p \in [p^*-\epsilon, p^*+\epsilon] \), use Taylor (\( \cdot R(p^*) = 0 \))

\[ R(p) \approx R(p^*) + C(p-p^*)^2, \quad C = R''(p^*) \]

For \( p \in [0, p^*-\epsilon] \cup [p^*+\epsilon, 1] \), since the set is compact and \( R(p) < R(p^*) \), we can find \( B_1, B_2 \) s.t. \( B_1(p-p^*)^2 \leq R(p^*) - R(p) \leq B_2(p-p^*)^2 \). Now take \( C = \min(B_1, B_2) \).
Lemma - \( \mu^* > \beta \rho^* \bar{F}(\rho^*) - C_2 / k^2 \)

**Pf** - \( \exists i \) s.t. \( |\rho^* - i/k| \leq 1/k \)

\( \Rightarrow [R(\rho^*) - R(i*/k)] \leq C_2 / k^2 \)

Lemma - Suppose we sort \( \Delta_i \) as \( \tilde{\Delta}_0 \leq \tilde{\Delta}_1 \leq \ldots \leq \tilde{\Delta}_k \)

Then \( \tilde{\Delta}_j \geq C_1 (j/2k)^2 \)

**Pf** - At most \( j \) elements of \( \{1/k, 2/k, \ldots, k/k^3 \} \) lie within distance \( j/2k \) of \( i*/k \).

Now consider \( \bar{R}_n - n/\mu^* \)

From UCB - \( -(\bar{R}_n - n\mu^*) \leq \sum_{i : \rho_i < \rho^*} \left( \frac{8 \log n}{\Delta_i} + 2 \right) \)

\( \leq \frac{32k^2}{C_1^2} \cdot \frac{n^2 \log n}{6} + 2k \)

\( = O(k^2 \log n) \)

On the other hand, \( nR(\rho^*) - \bar{R}_n \leq nC_2 = o(n/k^2) \)

Choosing \( K = (n / \log n)^{1/4} \) \( \Rightarrow nR(\rho^*) - \bar{R}_n \leq O(\sqrt{n \log n}) \)
However we do not know $n$.

- **Doubling trick**
  
  Use $n_0 = 1$, $n_1 = 2$, $n_2 = 4$, ..., $n_k = 2^k$.
  
  This continues till $2^{e^*} \leq n \Rightarrow e^* = \log_2 n$.
  
  However, regret $= \sum_{k=0}^{e^*} (Ne \log n)^{1/2} = \sum_{k=0}^{\log_2 n} (e 2^k)^{1/2}$
  
  $= O(\sqrt{n \log n})$

- Finally, we can also show $\overline{R}_n^{opt} - \overline{R}_n^* = O(\sqrt{n \log n})$.
  
  - This follows from Chernoff bounds. See KL'03.

- This result is also near-optimal.
  
  **Theorem (KL'03)** - No policy $\Pi$ can achieve $\overline{R}_n - \overline{R}_n^* = o(\sqrt{n})$.

- **Intuition** - Consider 2 coins of prob $1/2$, $1/2 + \varepsilon$.
  
  We need $\Omega(1/\varepsilon^2)$ trials to accurately identify the better coin.

  Now we can use this to construct a worst-case $F$s.t. $\text{regret} \geq \Omega(1/\varepsilon)$.