CMPUT 498/501: Advanced Algorithms **Fall 2025**

Lecture 24 (Dec 3, 2025): Prophet Inequality

Lecturer: Mohammad R. Salavatipour *Scribe: Lyndon Hallett, Haoxin Sang*

24.1 Secretary & Prophet

Recall the secretary problem from last lecture. We have n candidates for a secretary position and we assume there is a universal ordering of them, i.e. if we compare any two candidate i & j we can say which one is better for the job. The candidates arrive in a u.r. (uniformly random) order; each time we interview a candidate we must decide to either hire (and terminate the process) or pass and continue.

Goal: Find a strategy to maximize the prob. of hiring the best candidate.

We proposed the following algorithm:

- Pass for the first $\frac{n}{e}$ candidates.
- Then hire the candidate that is the best seen so far.

Theorem 1 *This algorithm finds/hires the best candidate with prob.* $\frac{1}{e}$.

We showed that if we pass k and then hire the best, we hire the best with prob. $(1-o(1))\frac{k}{n}\ln(\frac{n}{k})$ and with $k \approx n/e$ this is maximized.

We can show that in this problem the best strategy is in fact a wait-and-pick strategy like what we described.

Theorem 2 *The strategy that maximizes picking the best candidate is a wait-and-pick strategy.*

Proof. Consider some fixed optimal strategy. Let

- p_i: prob. the strategy picks agent at position i.
- q_i: prob. pick agent at position i conditioned on it being the best seen so far.

So $q_i = \frac{p_i}{1/i} = i \cdot p_i$.

$$\begin{aligned}
\text{Pr[Picking the best]} &= \sum Pr[\text{i'th agent is the best \& we pick it}] \\
&= \sum Pr[\text{i'th agent is the best }] \cdot q_i \\
&= \sum_i \frac{1}{n} q_i = \sum_i \frac{i}{n} p_i.
\end{aligned}$$

Clearly $0 \le p_i \le 1$. And we have:

$$\begin{aligned}
p_i &= Pr[\text{pick agent } i \mid i \text{ is best so far}] \cdot Pr[i \text{ is best so far}] \\
&\le Pr[\text{did not pick } 1, \ldots, i-1 \mid i \text{ is best so far}] \cdot \frac{1}{i}.
\end{aligned}$$

But not picking $1, \ldots, (i-1)$ is independent of i being the best so far. So $p_i \le \frac{1}{i}(1 - \sum_{j<i} p_j)$.

So we can upper-bound the success prob. of this (and any) strategy by the following LP:

$$\text{maximize } \sum_i \frac{i}{n} \cdot p_i$$

$$\text{Subject to:}$$

$$i \cdot p_i \le 1 - \sum_{j<i} p_j$$

$$0 \le p_i \le 1$$

It can be verified that:

$$\begin{cases} p_i = 0 & \text{if } i < k, \\ p_i = (k-1)\left(\frac{1}{i-1} - \frac{1}{i}\right) & \text{if } k \le i \le n, \end{cases}$$

is a feasible solution where k is the smallest integer s.t. $H_{n-1} - H_{k-1} \le 1$. It can be shown that the dual LP has the same value, and thus must be optimum.

We have a wait-and-pick strategy whose value is the same as this LP solution: For position i, if we have not picked any & this is the best so far, pick with prob.

$$\frac{ip_i}{1 - \sum_{j<i} p_j}.$$

Then it can be shown that this is picking the best candidate with prob. $\sum_i \frac{i}{n} p_i$ which is the same as the LP. ■

24.1.1 Extension: Multiple agents

Suppose we want to hire k agents (or pick k items) instead of just 1.

Goal: maximize the expected sum of the values of these k items.

Say the set of k largest items are $S^* \subseteq [n]$ and their total values is $V^* = \sum_{i \in S^*} a_i$.

First approach: Suppose we pass on the first $n/2$ agents. Note that we expect half $(k/2)$ of the largest items to be in the first half and half $(k/2)$ be in the second half. Suppose we use the value $\tilde{a}$ of the $(1-\epsilon)\frac{k}{2}$-th largest element in the first half as the threshold and pick up to k items from the second half if their values are $\ge \tilde{a}$.

We can formalize this: Let $\delta = O(\frac{\log k}{k^{1/3}})$ and $\epsilon = \frac{\delta}{2}$. Our goal is an algorithm that gives value $V^*(1-\delta)$ (Note $\to 1$ when $k \to \infty$). Pass the first $n/2$ items. Let $\tilde{a}$ be the value of $(1-\epsilon)\delta k$'th highest among the "passed" items; pick the first k elements in the remaining $(1-\delta)n$ with value $\ge \tilde{a}$.

Bad events:

- E_1: if $a' = \min_{i \in S^*} a_i$ is the lowest value of S^* and $\tilde{a} < a'$ (we pick items with lower value).
- E_2: # of items from S^* in the last $(1-\delta)n$ and greater than $\tilde{a}$ is much smaller than k.

Why these bad events happen rarely: For E_1, less than $(1-\epsilon)\delta k$ items from S^* fall in the first δn locations; this requires the # of them is smaller than $(1-\epsilon)\times$ expectation. Using Chernoff-Hoeffding this prob $\le e^{-\epsilon^2 \delta k} \approx \frac{1}{poly(k)}$.

For E_2 to happen, more than $(1-\epsilon)\delta k$ of the top $(1-\delta)k$ from S^* must be in the first δn elements. This means the # of them exceeds $(1+O(\epsilon))$ times the expectation, again has probability $\leq e^{-\epsilon^2 \delta k} = \frac{1}{poly(k)}$.

Best possible: [Kleinberg05] showed that we can get $1 - O(\sqrt{1/k})$ competitive ratio for the k secretary & this is best possible.

24.2 The Prophet Inequality

The setting and the goal of this is slightly different. We have a distribution for the value of each item but we don't know the actual value until the candidate is sampled (interviewed). The arrival of candidates/items is not random but adversarial!

Formally: An ordered set of random variables $X_1, \ldots, X_n$, with known distribution. At each step i, we sample $x_i \sim X_i$ and get its value: we can decide to accept (terminate) or pass (continue).

Goal: maximize the expected value of chosen item(s) compared to $\max_i X_i$. i.e. find an algorithm A s.t. $\mathbb{E}[A] \geq \alpha \cdot \mathbb{E}[\max_i X_i]$ while maximizing α.

First we show we cannot have $\alpha > \frac{1}{2}$. Suppose $X_1 = 1$ is always 1 while $X_2 = 0$ with prob $1-\epsilon$ and $X_2 = \frac{1}{\epsilon}$ with prob ϵ. Then $\mathbb{E}[\max\{X_1, X_2\}] = (1-\epsilon)\cdot 1 + \epsilon \cdot \frac{1}{\epsilon} = 2 - \epsilon$.

Any strategy either picks X_1 (which always has value 1) or picks X_2 and in either case $E[\text{algorithm}] = 1$. So we cannot have $\alpha > \frac{1}{2}$, but surprisingly we can get $\alpha = \frac{1}{2}$.

Theorem 3 ([KSG78]) *There is a strategy with $\alpha = \frac{1}{2}$.*

Proof. The idea is similar to the secretary problem, however because the arrival is not random (could be adversarial) we cannot ignore a fixed collection of values. Define $X_{\max} = \max_i X_i$ and let $\tau = \frac{1}{2}\mathbb{E}[X_{\max}]$. The algorithm is: Select the first x_i such that it is $\geq \tau$.

For any random variable Y let $Y^+ = \max\{Y, 0\}$. Then:

$$\begin{aligned}
\mathbb{E}[X_{\max}] &= \mathbb{E}[\tau + X_{\max} - \tau] \\
&= \tau + \mathbb{E}[(X_{\max} - \tau)^+] \qquad \text{since } (X_{\max} - \tau)^+ \geq X_{max} - \tau \\
&\leq \tau + \mathbb{E}[\sum_i (X_i - \tau)^+] \\
&= \tau + \sum_i \mathbb{E}[(X_i - \tau)^+] \qquad (*)
\end{aligned}$$

Let q be the probability that the algorithm returns some value $q = \Pr[\exists i : X_i \geq \tau] = 1 - Pr[\forall i : X_i < \tau]$.

Then we have:

$$\begin{aligned}
\mathbb{E}[\text{Alg}] &= \sum_i \underbrace{\Pr[\bigwedge_{j<i} (X_j < \tau) \wedge (X_i \geq \tau)]}_{\text{Alg}=X_i} \cdot \mathbb{E}[X_i \Big| \bigwedge_{j<i} (X_j < \tau) \wedge (X_i \geq \tau)] \\
&= \sum_i \Pr[\bigwedge_{j<i} (X_j < \tau)] \cdot \underbrace{\Pr[X_i \geq \tau] \cdot \mathbb{E}[\tau + (X_i - \tau)^+ | X_i \geq \tau]}_{\tau + \mathbb{E}[(X_i + \tau)^+]}
\end{aligned}$$

Note that $q = \sum_i \Pr[\bigwedge_{j<i}(X_j < \tau)] \cdot \Pr[X_i \geq \tau]$.

Finally, we have:

$$\begin{aligned}
\mathbb{E}[\text{Alg}] &= q \cdot \tau + \sum_i \Pr[\bigwedge_{j<i}(X_j < \tau)] \cdot \mathbb{E}[(X_i - \tau)^+] \\
&\geq q \cdot \tau + \sum_i (1-q)\mathbb{E}[(X_i - \tau)^+] \\
&\geq q \cdot \tau + (1-q)(\mathbb{E}[X_{\max}] - \tau) \quad \text{using (*)} \\
&= \tau.
\end{aligned}$$

■

24.2.1 LP-based Generalization

The following LP-based approach works for more general settings of prophet inequalities. Suppose X_i comes from a known distribution and say it will have value $v_i \geq 0$ with probability p_i and is zero otherwise (for simplicity we are focusing on disc. version but the same argument works in general; in general, we can have: $\Pr[X_i \geq v_i] = p_i$).

Consider the following LP:

$$\begin{aligned}
\text{maximize:} \quad & \sum_i x_i \cdot v_i \\
\text{subject to:} \quad & x_i \leq p_i \qquad \forall i \\
& \sum_i x_i \leq 1 \\
& x_i \geq 0
\end{aligned}$$

Let x^* be the optimal LP solution.

Lemma 1 $\sum_i x_i^* \cdot v_i \geq \mathbb{E}[X_{\max}]$

Proof. We give a feasible sol with objective value $\mathbb{E}[X_{\max}]$. Let q_i be the prob. that X_i is non-zero and is X_{max}. Note $q_i \leq p_i$ and $\sum q_i \leq 1$. Also $\sum_i q_i \cdot v_i = \mathbb{E}[X_{max}]$. ■

Using this lemma we can design a $\frac{1}{4}$-competitive alg.

Idea: The LP solution x^* suggests how frequently we accept X_i (x_i^*). If we had no constraints on how many X_i's we can accept we could say: accept X_i when it has value v_i with probability $\frac{x_i^*}{p_i}$, unless we have accepted something.

This could have poor performance: e.g. $X_1 = 1$ w.p. 1, and

$$x_2 = \begin{cases} \frac{1}{\epsilon^2} & \text{w.p.} \quad \epsilon, \\ 0 & \text{otherwise.} \end{cases}$$

We have $x^* = (1-\epsilon, \epsilon)$, but an algorithm that always accepts X_1 w.p. $1-\epsilon$ reaches X_2 without accepting anything at most ϵ-fraction of the time. So

$$\Pr[\text{reach } X_2] \cdot \Pr[X_2 = \frac{1}{\epsilon^2}] = \epsilon^2.$$

Therefore, expected value is at most $(1-\epsilon)+\epsilon^2 \cdot \frac{1}{\epsilon}^2 = 2-\epsilon$ but $\mathbb{E}[X_{\max}] = \frac{1}{\epsilon}$.

What if we ignore each element we reach with probability $\frac{1}{2}$? (i.e. we don't even look at it!)

Algorithm: Let x^* be LP solution. When looking at item i (assuming nothing is selected), ignore it with prob $\frac{1}{2}$ (without even looking), else accept with prob x_i.

Lemma 2 *This algorithm achieves a value* $\frac{1}{4}\mathbb{E}[X_{max}]$.

Proof. We say that we reach item i if we have not picked any item before. We pick each i with prob at most $\frac{x_i^*}{2}$ (1/2 ignore it and pick w.p. x^* if we reeach i)

$$
\begin{aligned}
E[\text{Alg}] &\geq \sum_{i=1}^{n} \Pr[\text{reach } i] \cdot \frac{1}{2} \cdot x_i^* \cdot v_i \\
&\geq \sum_{i=1}^{n} Pr[\bigwedge_{j<i} \text{Not picked } j] \cdot \frac{x_i^* v_i}{2} \\
&\geq \sum_{i=1}^{n} (1 - \sum_{j<i} \frac{x_j^*}{2}) \frac{x_i^* v_i}{2} \\
&\geq \sum_{i=1}^{n} \frac{1}{2} \cdot \frac{x_i^* v_i}{2} \qquad \text{since } \sum_j \frac{x_j^*}{2} \leq \frac{1}{2} \\
&\geq \frac{1}{4}\mathbb{E}[X_{\max}]
\end{aligned}
$$

■

24.3 Reference

[Kleinberg05] Kleinberg, Robert D. A multiple-choice secretary algorithm with applications to online auctions. In SODA, vol. 5, pp. 630-631. 2005.

[KSG78] U. Krengel and L. Sucheston. On semiamarts, amarts, and processes with finite value. *Probability on Banach spaces*, 4:197–266, 1978.