CSC2421: Online and other myopic algorithms Spring 2021

Allan Borodin

March 11, 2021

Week 8

Agenda for today and following two weeks

- For the last two meetings, we had overview presentations of the reading projects. I am basing the grade on the quality of these presentations. The criteria are : Was the presentation clear and how informative was the presentation.
- Koosha and Xiaoxu will give their more detailed presentations next week March 18. Alex, Jinman and Koko will follow on March 25. If you cannot prepare sufficiently by those dates please let me know. I am allowing approximately 50 minutes for the next presentations.
- During the weeks when there are no presentations (including today), I will present some other topics in some detail.

Todays agenda

- Today we will discus three related problems: the secretary problem, prophet inequalities, and the prophet secretary problem.
- These all can be considered within the framework of online bipartite matching.
- The prophet secretary results are relatively new and based mainly on papers by Esfandiari et al [SICOMP 2017], Ehsani et al [SODA 2018] and Correa et al [Math Programming 2020].
- The secretary and prophet inequalities results are more "classic". The first published algorithm and proof for the secretary problem is attribued to Lindley (1961). The prophet inequalities bounds is due to Krengel and Sucheston (1977) whereas the threshold algoriehm presented here is due to Samuel-Cahn (1984) using the proof by Kleinberg and Weinberg (2012) as presented in Lucier (2017).

The random order model

Aa we have mentioned, worst case analysis can often be misleading, as the bounds can be too pessimistic and at odds with the "real world" performance of some online and other conceptually simple algorithms.

The basic online model and what was initially *competitive analysis* was a game between the online algorithm and an adversary. For deterministic and randomized online algorithms (wrt oblivious adversaries), an adversary sees the algorithm and creates a nemesis input set $\{I_1, I_2, \ldots, I_n\}$ and an adversarially chosen input sequence of the input set.

In the random order model, the adversary still creates a nemesis input set but then the sequence of input items arrives in random order. That is, in the random order model (ROM), the inputs arrive in the order $\pi(1), \pi(2), \ldots, \pi(n)$ where π is chosen uniformly at random from the $n!$ possible orderings.
We already saw (Week 4) that any deterministic online greedy algorithm for unweighted maximum bipartite matching has competitive ratio $1/2$ while in ROM, the natural greedy algorithm has ratio $1 - 1/e$.

The random model continued

I will continue to view the random order model (and other stochastic models which involve distributions on input items) as online models (and continue to use the terminology of "competitive ratio") since the online algorithm has no control over the input sequence.
In the ROM, we often say that "nature" selects the permutation.

Whenever we have some stochastic aspect to the input sequence, the algorithm's performance is the expected value the algorithm achieves for the objective function.

Since the benchmark is the optimal value for the input set, the benchmark in the random order model is not impacted by the random order and hence we do not need to take an expectation for OPT.

When the input set or sequence comes from a distribution, the usual benchmark is to take the expectation of OPT with respect to the randomness in the input.

The secretary problem

As I am sure I have mentioned before, to the best of my knowledege, the random order model was first used in the secretary problem. The terminology may not be politically correct. The problem has also been called the "dowry problem" which is more problematic. It might be best to call this the candidate problem but the name secretary problem has stuck.

The problem is to select one candidate (one item) from a worst case set of n candidate values. Another very useful interpretation is for a seller to select one buyer who wishes to purchase an item.

Each candidate arrives online and (say by interviewing) we learn the value of this candidate. For each candidate, the algorithm (i.e. the interviewer) must either irrevocably reject the candidate or irrevocably accept the candidate (in which case, the algorithm stops).

This then can be seen as a special case of edge or vertex weighted online bipartite matching where there is one single offline vertex.

The secretary problem continued

Since the candidate values are completely adversarial, it is easy to see that there cannot be any competitive ratio not depending on the values if the order of arrivals is adversarial. (This is true even for $n = 2$ candidates.)

But in ROM, the optimal competitive ratio is $\frac{1}{e}$. In fact, the solution to the secretary problem is an algorithm that (as $n \to \infty$) selects the best candidate (i.e., the highest value) with probability $\frac{1}{e}$ which implies that the expected value of the algorithm is at least $\frac{1}{e}$.

It can be shown that this is the optimal expected value.

Unlike most (or maybe all) of the online problems we have considered, here we assume that the algorithm knows the number n of candidates.

The secretary algorithm

An historical account for the secretary problem is given by Ferguson (1989) and he attributes the first solution to Lindley (1961).
Note: I think TCS interest in ROM begins with the KVV (1990) algorithm for online bipartite matching. Here is the secretary algorithm copied from the text.

Algorithm 44 A deterministic algorithm for the Secretary problem in ROM.

```
procedure Secretary
    v_best ← v_1
    i ← 2
    r ← ⌊n/e⌋
    while i ≤ r do                  ▷ Find the best candidate amongst the first r = n/e candidates
        if v_i > v_best then
            v_best ← i
    i ← i + 1
    while i ≤ n do                  ▷ Output next candidate (if any) better than initial best
        if v_i > v_best then return i
```

Proof of the secretary bound

Intuitively and informally, for any choice of an initial set of $r = r(n)$ candidates, there is a probability that the second best will occur amongst the first r candidates while the best candidate will occur after the first r candndiates.

It is easy to see that for any choice of $c < 1$, $r = cn$ will result in a constant competitive ratio. For example, if $c = 1/4$, we immediately get *at least* a comeptitive ratio $3/16 \approx .1875 < 1/e \approx .3678$.

But of course there are other ways that the best candidate could be found. For example when $c = 1/4$, it could be that the third best canddiate is in the first r candidates and the while the best precedes the second best and occurs after the first r candidates, etc. etc,

Enumerating all the possibilities and their probability of occurence would be painfully tedious at best and really a combinatorial nightmare. Furthermore we are interested in this probability as $n \to \infty$.

The secretary bound continued

Instead by estimating probabilities using continuous analysis, the calculation $\mathrm{f}r(n) = cn$ becomes manageable and one can then optimize for the value of c. As stated in the secretary algorithm, the optimum value of $c = 1/e$ and $r(n) = \lfloor n/e \rfloor$.

I will go over the proof as we state it in the text. We can assume that all values are distinct since identical values only make it more likely to select the best value. Then the necessary and sufficient conditions for the algorithm to select the best candidate:

- The maximum value occurs in position $t+1$ for some $t \geq r(n)$ which occurs with probability $1/n$. And
- The maximum value of the first t candidates is the same as the maximum value of the first $r(n)$ candidates which occurs with probability $r(n)/t$

Summing up over all $t \geq r(n)$, the desired probability is $p_c(n) = \sum_{t=r(n)}^{n-1} \frac{r(n)}{t} \frac{1}{n} = \frac{r(n)}{n} \sum_{t=r(n)}^{n-1} \frac{1}{t}$

Finishing the proof

Claim: In the limit as $n \to \infty$, we have

$$\sum_{t=r(n)}^{n-1} \frac{1}{t} = \int_c^1 \frac{1}{x} dx = -\int_1^c \frac{1}{x} dx = -\ln c$$

So the probability for $r(n) = cn$ (in the limit) is

$$p_c(n) = \frac{cn}{n} \sum_{t=r(n)}^{n-1} \frac{1}{t} = -c \ln c$$

which is optimized at $c = \frac{1}{e}$.

Beyond worse case: inputs generated from distributions

Stochastic optimization usually refers to optimization problems where we assume that input sets are not fully specified but rather are specified by distributions. In stochastic optimization, the outcome of an execution of an algorithm becomes a random process where inputs are instantiated as random values drawn from the distribution. For randomized algorithms, the process is then in terms of both the randomness in the input as well as the randomness in the algorithm.

And what is the benchmark? We assume a non-adaptive adversary that initially chooses the distribution $\mathcal{D}$ for the input set. The input sequence can then either be adversarial or random order. We will adopt the usual convention that the benchmark is the expectation (over the input instantiations) of an optimal solution for each instantiation. That is, the competitive ratio is defined as:

$$\max_{\mathcal{D}} \frac{\mathbb{E}_{\mathbf{x} \in \mathcal{D}}[ALG(\mathbf{x})]}{\mathbb{E}_{\mathbf{x} \in \mathcal{D}}[OPT(\mathbf{x})]}$$

Independence

A common (but often unrealistic) assumption is that the distributions $\mathcal{D}$ are product distributions, that is, input items are defined by *independent distributions* and we refer to such a process as as an "i.d. process". That is, each input item instance w_i is drawn independently from a distribution X_i.

A further assumption is that the distributions are identical and we refer to such a process as an "i.i.d. process".
As stated, independence is often but not always unrealistic. Currently in TCS, i.d. and i.i.d. processes are still the most common assumption although as we have seen (e.g., in Markov paging), where we assume that the input item are correlated.

Question: Is independence a reasonable assumption for the secretary problem? That is, is it reasonable to assume that the values each of each candidate (or buyer) are are determined by indepedent distributions?

Adopting the distributional assumption to the secretary problem

The prophet inequalities and prophet secretary problems are i.d. distributional analogues of the secretary problem. We assume the algorithm is given n input item distributions $X_1, X_2, \ldots, X_n$ from which values will be drawn.

In the prophet inequalities problem, the input sequence is an adversarially defined sequence $(X_{\pi(1)}, w_{\pi(1}), (X_{\pi(2)}, w_{\pi_(2)}), \ldots, (X_{\{\pi(n)}, w_{\pi(n)})$ where π is determined by the adversary.

We adopt the convention that the algorithm knows the set of distributions but not the order which is being revealed one input item at a time. In the prophet secretary problem, the sequence order is determined randomly; that is, π is chosen uniformly at random.

Threshold algorithms for the prophet inequalities and prophet secretary problems.

In the prophet problems, upon seeing an input item $(X_{\pi(i)}, w_{\pi_(i)})$, the algorithm must irrevocably decide whether to take the item (and then the process stops) or reject the item and go on to the next item $(X_{\pi(i+1)}, w_{\pi(i+1)})$ For $i = n$, the algorithm might as well accept the last item but results do not usally depend on taking the last item.

This decision as to accepting or rejecting an item $(X_{\pi(i)}, w_{\pi(i)})$ can be seen to be equivalent to setting a threshold $\tau_{\pi(i})$ and accepting if and only if $\tau_{\pi(i)} \leq w_{\pi(i)}$. In general, the threshold $\tau_{\pi(i)}$ can be a function of $i, X_{\pi(i)}, X_{\pi(j)}$ (for all $j > i$).

Implicitly, when seeing the input $(X_{\pi(i)}, w_{\pi(i)})$, the algorithm knows that the first $j < i$ items were rejected and knows what distributions remain.

Non-adaptive threshold algorithms are a restricted class of threshold algorithms where each τ_i as a function of just X_i and $\mathcal{D}$. That is, the algorithm has no knowledge of the history thus far nor what input items still remain to be seen.

The non-adaptive threshold template

Algorithm 48 The non-adaptive thresholds template for prophet inequalities and prophet secretaries.

procedure *Prophets*

  $\pi : [1, n] \to [1, n]$ is a permutation ▷ For prophet inequalities, π is adversarially chosen; for prophet secretaries, π is a random permutation.

  $\tau_1, \tau_2, \ldots, \tau_n$ is a sequence of thresholds ▷ The thresholds are chosen by the algorithm based on the set of distributions before the individual online items arrive. In particular, the thresholds do not depend on when an item arrives.

  $flag \leftarrow 0$

  for $j = 1..n$ and while $flag = 0$ **do**

    Draw $w_{\pi(j)}$ from distribution $X_{\pi(j)}$

    if $j < n$ and $w_{\pi(j)} \geq \tau_{\pi(j)}$ **then**

      Accept the $\pi(j)$ item

      $flag \leftarrow 1$

    else if $j = n$ **then**

      Accept the $\pi(n)$ item

An optimal single threshold algorithm for prophet inequalities

It turns out that by restricting items to come from a known set of independent distributions, we can achieve a much improve competitive ratio than the optimal ratio for the worst case secretary problem (even though the input order is adversarial).

Moreover, there is a simple non-adaptive *single threshold* stopping rule. Let $T = \frac{\mathbb{E}[\max_i w_i]}{2}$. The uniform stopping rule is to accept the first $w_i \geq T$ and then stop. This threshold stopping rule turns out to be the best online algorithm.

Theorem

Let $\mathcal{A}$ be the online algorithm using the above stopping rule. For every set of independent distributions $X_1, X_2, \ldots, X_n$, $\mathcal{A}$ has competitive ratio at least $\frac{1}{2}$. Furthermore this is an optimal stopping rule in the sense that for any $\epsilon > 0$, there are distributions such that the expected value of any online algorithm is at most $(\frac{1}{2} - \epsilon) \cdot \mathbb{E}[max_i w_i]$.

Proof of threshold algorithm for prophet inequalities

Proof

- For the negative example, let X_1 be a deterministic distribution with value 1 and let X_2 be the distribution such that $w_2 = \frac{1}{\epsilon}$ with probility ϵ and 0 with probability $1 - \epsilon$ for some arbitrarily small $\epsilon > 0$. An online algorithm can accept $w_1 = 1$ and obtain that value or it can decide to reject w_1 and then obtain $\mathbb{E}[X_2] = 1$. So that any online algorithm will obtain expected value 1. On the other hand.

  Claim For the above distributions, $\mathbb{E}[\max\{X_1, X_2\}] = (2 - \epsilon)$. While the claim may seem surprising at first, the result follows from a simple conditional expectation argument. Let Y be any random variable and let Z be a random $\{0, 1\}$ indicator variable with $p = Prob[Z]$. Then

$$\mathbb{E}[Y] = \mathbb{E}[Y|Z] \cdot p + \mathbb{E}[Y|\bar{Z}] \cdot (1 - p)$$

  The claim follows by setting $Y = \max\{X_1, X_2\}$, and letting Z be the indicator variable for $X_1 \geq X_2$ so that $p = prob[Z] = (1 - \epsilon)$.

Proof of threshold algorithm continued

We now consider the positive result showing that the algorithm achieves the desired competitive ratio.
We will use an economic interpretation for the analysys of the single threhold algorithm. Namely, we will interpret the threshold as a price that the seller is offering to each buyer and that price becomes the sellers revenue if any buyer accepts an offer.

The utility of a successful buyer is then the value to the buyer minus the price. We want to find lower bounds for $\mathbb{E}[\mathit{Revenue}]$ and $\mathbb{E}[\mathit{Utility}]$.

The desired objective is then $\mathbb{E}[ALG_\tau] = \mathbb{E}[\mathit{Revenue}] + \mathbb{E}[\mathit{Utility}]$. In economic terminology the objective is called the *social welfare*.

Note: There is a similar economic presentation of the KVV Ranking bipartite matching algorithm.

Proof of threshold algorithm continued

- The revenue to the seller is precisely the threshold T times the probability that there is a successful sale. That is,

$$\mathbb{E}[Revenue] = T \cdot \Pr[\text{the item is sold}] = \frac{1}{2} \max_i X_i \cdot \Pr[\text{the item is sold}]$$

- To calculate the expected utility, we consider the expected utliltiy $\mathbb{E}[Utiltiy_i]$ of each buyer i. Only one buyer can be successful and the purchase by buyer i only depends on the value w_i and the event that that the item has not been previously sold to a previous buyer $j < i$, an event which is independent of w_i . Therefore, $\mathbb{E} = \sum_i \mathbb{E}[Utiltiy_i]$.

Letting $(w_i - T)^+ = \max(w_i - T, 0)$ we have :

$$\mathbb{E}[Uiltity_i] = \mathbb{E}[(w_i - T)^+] \cdot \Pr[\text{item was not sold to any buyer } j < i]$$

and thus

$$\mathbb{E}[Uiltity] = \sum_i \mathbb{E}[(w_i - T)^+] \cdot \Pr[\text{item was not sold to any buyer }]$$

End of proof of threshold algorithm

Finally,

$$\sum_i \mathbb{E}[(w_i - T)^+] \geq \mathbb{E}[\max_i (w_i - T)^+] \geq \mathbb{E}[\max_i w_i] - T = \mathbb{E}[\max_i X_i]/2$$

so that

$$\begin{aligned}\mathbb{E}[\mathcal{A}] &= \mathbb{E}[\mathit{Revenue}] + \mathbb{E}[\mathit{Utility}] \\ &\geq \tfrac{1}{2}\mathbb{E}[\max_i X_i] \cdot \Pr[\text{item was sold}] + \tfrac{1}{2}\mathbb{E}[\max_i X_i] \cdot \Pr[\text{item was not sold}] \\ &= \tfrac{1}{2}\mathbb{E}[\max_i X_i]\end{aligned}$$

End of Proof

Comments on the theorem and proof

Note that the negative result holds even when the algorithm knows the order X_1, X_2 of the input sequence while the positive result holds without knowledge of the order.

In fact, the positive results holds if only $OPT = \mathbb{E}[\max_i X_i]$ is known. Furthermore if the algorithm could control the order and reveal (X_2, w_2) first, then the algorithm would be optimal for this set of distributions.

This begs the question as to what if the input order was random?

It also begs the question as to finding a "best order" which we postpone for now as that takes us beyond the online framework.

The prophet secretary problem

As already defined the prophet secretary problem assumes that there is a known set of distributions $\mathcal{D} = \{X_1, \ldots, X_n\}$ given by an adversary and the input item $(X_{\pi(i)}, w_{\pi(i)})$ arrive in random order.

Theorem

There is a single threshold (and hence non-adaptive) algorithm for the prophet secretary problem with competitive ratio $1 - \frac{1}{e}$. This is the optimal competitive ratio for non-adaptive thresholds.
Note: *The algorithm is a deterministic algorithm for continuous distributions but requires randomization to "break ties" when the single threshold $\tau = w_j$ for some j. More specifically, without tie breaking no deterministic single threshold algorithm can achieve a competitive ratio asymptotically better than $\frac{1}{2}$.*

A bad example for a deterministic single threshold algorithm

We first note that for a continuous distribution, and no matter how many distinct thresholds are used, the probability of a tie is infinitessimally small. But now consider the following discrete probability example:
There are $n-1$ deterministic distributions such that $\Pr[X_i = 1] = 1$ and one distribution X_i such that $\Pr[X_i = n] = 1/n$ and 0 otherwise. It follows that:

1. For any fixed threshold $\tau < 1$, the algorithm obtains the value n with probability $1/n^2$ and 1 otherwise so that that algorithm's expected value approaches 1 as $n \to \infty$,
2. For any fixed threshold $\tau \geq 1$, the algorithm obtains value n with probability $1/n$ (and 0 otherwise) and therefore the algorithms expected value is precisely 1.

Hence in either case, the algorithm's expected value is 1 as $n \to \infty$. It can can shown that $\mathbb{E}[\max_i X_i] \to 2$ as $n \to \infty$ so that the asymptotic competitive ratio is $\frac{1}{2}$ for the given distributions.

Motivating the use of tie-breaking for randomization

If we randomize the decision for when $w_{\pi(j)} = \tau_j$, it turns out that there is a single threshold that provides a $1 - \frac{1}{e}$ competitive ratio for the prophet secretary problem.

For the example used in the comment above, if the algorithm only accepts the deterministic distribution with probability $\frac{1}{n}$, the expected value for the algorithm becomes $1 + \frac{1}{e}$ as $n \to \infty$ so that the asymptotic ratio for this set of distributions is $\frac{1+\frac{1}{e}}{2} \approx .6843$.

Intuitively, we are choosing the probability inversely proportional to the expectation of obtaining a tie and hence we are likely to accept the large value n (if it is drawn) and also still likely to accept the deterministic value of 1 if the large value is not drawn.

The proof for the fixed threshold prophet secretary algorithm with competitve ratio $1 - \frac{1}{e}$

We will assume continuous distributions so as to ignore the tie-breaking needed for non-continuous (e.g. discrete) distributions.

We set the threshold τ so that $\Pr[\max_i X_i \leq \tau] = \frac{1}{e}$.

We will again use the economic interpretation for the analysys of the single threhold algorithm. Namely, we will interpret the threshold as a price that the seller is offering to each buyer and that price becomes the sellers revenue if any buyer accepts an offer.

Namely, the desired objective is $\mathbb{E}[ALG_\tau] = \mathbb{E}[Revenue] + \mathbb{E}[Utility]$.

Continuing the proof for the fixed threshold prophet secretary algorithm

We first bound $\mathbb{E}[\mathit{Revenue}]$. By the definition of τ, values below the threshold are drawn with probability less than $\frac{1}{e}$ so that the probability of a successful sale is $1 - \frac{1}{e}$. It follows that
$\mathbb{E}[\mathit{Revenue}] \geq (1 - \frac{1}{e}) \cdot \tau \geq (1 - \frac{1}{e}) \cdot \mathbb{E}[\mathit{OPT} \cdot \mathbf{1}_{\mathit{OPT}<\tau}]$

We will just state the following lemma that we need to bound $\mathbb{E}[\mathit{Utility}]$:

Lemma

Let $q(j) = \Pr[\max_{1 \leq k \leq j}[w_{\pi(k)} < \tau]$ and $q_{-i}(j) = \Pr[$The algorithm does not choose any of the first $j-1$ items $\mid \pi(j) = i]$. That is, $q(j)$ is the probability that the first j items were not chosen and $q_{-i}(j)$ is the probability that the first $j-1$ items were not chosen conditioned on the j^{th} item being X_i. Then $q_{-i}(j) \leq q(j)$.

Bounding the Utility of the buyer

Let u_i be the utility of buyer i and let $\mathbf{1}_{w_i \geq \tau}$ denote the indicator function specifying that $w_i \geq \tau$. Using the inequality from Lemma **??** we obtain:

$$
\begin{aligned}
\mathbb{E}[utiliy] = \sum_{i=1}^{n} \mathbb{E}[u_i] &= \sum_{i=1}^{n} \sum_{j=1}^{n} \Pr[\pi(j) = i] \cdot q_{-i}(j) \cdot \mathbb{E}[w_i \mathbf{1}_{w_i \geq \tau}] \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} q_{-i}(j) \cdot \mathbb{E}[w_i \mathbf{1}_{w_i \geq \tau}] \cdot \frac{1}{n} \\
&\geq \sum_{i=1}^{n} \sum_{j=1}^{n} q(j) \cdot \mathbb{E}[w_i \mathbf{1}_{w_i \geq \tau}] \cdot \frac{1}{n} \\
&= \sum_{i=1}^{n} \mathbb{E}[w_i \mathbf{1}_{w_i \geq \tau}] \sum_{j=1}^{n} q(j) \frac{1}{n} \\
&= \mathbb{E}\Big[\sum_{i=1}^{n} w_i \mathbf{1}_{w_i \geq \tau} \Big] \sum_{j=1}^{n} q(j) \frac{1}{n}
\end{aligned}
$$

Continuing the Utility bound

$\mathbb{E}\Big[\sum_{i=1}^{n} w_i \mathbf{1}_{w_i \geq \tau}\Big] \sum_{j=1}^{n} q(j)\frac{1}{n} \geq \mathbb{E}\Big[\sum_{i=1}^{n} OPT \cdot \mathbf{1}_{OPT \geq \tau}\Big] \sum_{j=1}^{n} q(j)\frac{1}{n}$

We then only need to bound $\sum_{j=1}^{n} q(j)\frac{1}{n}$ and that bound will be $1-\frac{1}{e}$

We this by first showing $q(j \geq exp\big(-\frac{j}{n}$ and then the summation is

.

- Let $p(i) = \Pr[w_i < \tau]$ and (as usual) $exp(z) = e^z$. For every j
$q(j) = \Pr\Big[\max_{1 \leq k \leq j}\{w_{\pi(k)} < \tau\}\Big] = \mathbb{E}_{\pi}\big[\prod_{k=1}^{j} p(\pi(k))\big]$
$= \mathbb{E}_{\pi}\Big[exp\big(\sum_{k=1}^{j} \ln p(\pi(k))\Big] \geq exp\Big(\mathbb{E}_{\pi}\Big[\sum_{k=1}^{j} \ln p(\pi(k))\Big]\Big)$
$= exp\Big(\frac{j}{n}\sum_{k=1}^{j} \ln p(\pi(k)\Big)$
$= exp\big(-\frac{j}{n}\big)$
where the inequality is due to the convexity of the function *exp* and the final equality follows from the definition of τ.
- For the desired summation we have
$\sum q(j) \cdot \frac{1}{n} \geq \sum exp(-\frac{j}{n}) \cdot \frac{1}{n} \geq \int_0^1 exp(-x)dx = (1-\frac{1}{e})$

Finishing up the proof of the competitive ratio for the fixed threshold prophet secretary bound

We have just concluded that
$\mathbb{E}(Utility) \geq \mathbb{E} \cdot \left[\sum_{i=1}^{n} OPT \cdot \mathbf{1}_{OPT \geq \tau} \right] \sum_{j=1}^{n} q(j) \frac{1}{n}$
$\geq \mathbb{E}\left[\sum_{i=1}^{n} OPT \cdot \mathbf{1}_{OPT \geq \tau} \right] \cdot (1 - \frac{1}{e}).$

Summing up, we obtain
$\mathbb{E}[ALG]_\tau = \mathbb{E}(Revenue) + \mathbb{E}(Utility) \geq$
$\mathbb{E}[OPT \cdot \mathbf{1}_{OPT < \tau}](1 - \frac{1}{e}) + \mathbb{E}\left[\sum_{i=1}^{n} OPT \cdot \mathbf{1}_{OPT \geq \tau} \right](1 - \frac{1}{e})$

Beating the $1 - \frac{1}{e}$ competitive ratio.

Can the $1 - \frac{1}{e}$ "barrier" be overcome? It turns out that there is a randomized *blind multi-threshold strategy* that can achieve an improved ratio. A blind strategy is a "slightly" adaptive threshold algorithm where the $\{\tau_i\}$ are chosen as follows:

- Let $\alpha : [0, 1] \to [0, 1]$ be a non-increasing function.
- Let $u_1, u_2, \ldots, u_n$ be drawn independently from $[0, 1]$
- Sort the $\{u_i\}$ so that $u_1 \leq u_2 \ldots \leq u_n$
- Choose τ_i such that $\Pr[\max_i X_i \leq \tau_i] = \alpha(u_i)$

The state of the art for the prophet secretary problem

We state the following theorem without proof. While the format of a blind stratregy is conceptually simple, the analysis is more involved than the analysis for the fixed threshold strategy.

Theorem

For the prophet secretary problem, we have have the following results:

- *For any set of distributions* $\{X_i\}$*, there is a blind threshold strategy with competitive ratio* .669
- *Every blind strategy has competitivee ratio at most .675*
- *Every online algorithm has competitve ratio at most* $\sqrt{3} - 1 \approx .732$.