1 Introduction

In this lecture we will be looking into the secretary problem, and an interesting variation of the problem called the incentive comparable secretary problem. For both of these problems we will show how they can be solved though a correspondence between all possible strategies and the feasible solution space to a linear program. In both cases this will allow us to derive an algorithm and prove that it is the best algorithm for the problem.

We will give an algorithm that is non-adaptive and must commit to box selection before moving on that achieves at least $\frac{1}{8}$ of the reward of the best adaptive algorithm even when the adaptive algorithm can choose the best box after they are all opened. The methods for solving this problem will be similar to the secretary problem, showing a correspondence between strategies of opt and the feasible solution space of a linear program. We will then use the optimal solution of the linear program as a bound for opt and show that we are competitive with this bound.

2 The Secretary Problem

We will now consider the classic secretary Problem. You are tasked with hiring a new secretary. You are given a list of $n$ applicants and tasked with interviewing them with the goal to hire the best applicant. Your company only wants the best applicant, and choosing the second-best will be considered a failure. Additionally, you must make your decision as to whether or not to hire a specific applicant immediately following the interview.

Note that in the adversarial setting this is completely hopeless. We cannot know any information about the best secretary and thus we can always be beaten by the adversary. For this reason we analyze the algorithm in the stochastic setting. That is, the adversary can choose the secretaries but the order is random. We must choose a strategy to maximize the chance of finding the best applicant, given adversarially chosen applicants in a randomly chosen order.

First we will consider a simple strategy that gives at least a $\frac{1}{4}$ chance at success. Simply look at the first $\frac{n}{2}$ applicants and reject them all, then choose the next applicant you see which is better than anything that comes before. Clearly if the second-best secretary is in the first half and the best is the the second half this algorithm hires the best secretary. Since the second-best secretary is in the first half with roughly $\frac{1}{2}$ probability and the best secretary is in the second half with roughly $\frac{1}{2}$ probability this algorithm succeeds with probability about $\frac{1}{4}$.

But we can do much better than this. In fact, the best strategy is the above algorithm with a cutoff at $\frac{n}{e}$ instead of $\frac{n}{2}$. This gives a success probability of $\frac{1}{e}$. Later today we will prove this result, and prove that this result is optimal by constructing and $LP$. 

An interesting modification of this problem is the incentive comparable secretary Problem. In this variant, we want to achieve the same goal as the classical secretary problem, but in a way that does not give a higher chance of getting hired to secretaries that arrive later. That is, so that any secretary, knowing our strategy for hiring, will not be incentivized to choose any specific position in the ordering over any other. This necessarily means that we will accept a sub-optimal secretary sometimes to make the probabilities even. Thus the protocol must be randomized. We can show
that the optimal such protocol will give a $1 - \frac{1}{\sqrt{2}}$ chance for every secretary and we will show that this is tight with an LP proof.

Finally there are many other versions of this problem that have been studied. The ones that we will discuss here and many others are studied in detail by Buchbinder et al. [2].

2.1 The Secretary Problem as a LP

We claim that we can describe any possible solution to the secretary problem as a solution to the following linear program:

Let $x_i = P$ (selecting the secretary at position $i$)

Here the probability is taken over the execution of the algorithm over all possible $n!$ input orderings.

We will show that any valid strategy for the secretary problem must satisfy the following LP. Hence, the LP below is a relaxation. It will also turn out that any feasible solution to this LP can be converted into a valid secretary algorithm without any loss in the objective. Thus, this LP captures the problem exactly.

Maximize: $\sum_{i=1}^{n} \frac{i}{n} x_i$

s.t. $\forall i, \sum_{j<i} x_j \leq 1 - \sum_{j<i} x_j$

$x_i \in [0, 1] \quad \forall i$

Feasibility of LP: We first show that any secretary strategy gives a feasible solution to this LP, and the LP objective exactly measures the expected objective value of the strategy.

The following is a key observation. At any time step $i$, even if we condition on which secretaries arrive during the first $i$ steps, since the order is random, the $i$-th secretary is equally likely to have rank $1, \ldots, i$ among the first $i$ secretaries. Moreover, the only information that the algorithm has at any time is the relative ranking of the secretaries it has seen thus far. Finally, since the algorithm only gets value if it picks the overall best secretary, at any step $i$, any algorithm should only bother picking a secretary (if at all) if it is the best among the ones seen so far.

Thus the game can simply be viewed as that at any step $i$, a secretary arrives which has rank (thus far) equally distributed among $1, \ldots, i$. These observations imply that a secretary strategy can be completely specified by describing the probability $q_i$ of picking the secretary at the $i$-th given that it is the best among the ones seen thus far.

So valid constraint is the following: The probability that secretary $i$ is picked given that it is best so far plus the probability that some secretary was already picked earlier is at most 1.

$P(i \text{ picked } | \text{ best so far}) \leq (1 - \sum_{j<i} x_j)$

As the probability that secretary $i$ is the best so far is exactly $1/i$.

$P(i \text{ is picked}) = P(i \text{ picked } | \text{ best so far})P(\text{best so far})$
and thus

\[ x_i \leq (1 - \sum_{j<i} x_j) \frac{1}{i} \]

Finally, to see the objective function, secretary \( i \) is the overall best with probability \( 1/n \). Moreover, in this case it is also the best among the first \( i \), and hence the algorithm will choose it probability \( q_i = ix_i \).

**LP to Secretary:** We have shown that algorithms for the secretary problem correspond to possible solutions for this linear program. However, it is still not clear that a solution to this Linear Program would be a solution to the secretary problem.

Given any \( x_1, x_2, \ldots, x_n \) that satisfy that LP constraints we can create an algorithm that can achieves the value of the function we are trying to maximize.

**Algorithm 1**

1: \( \text{for } i = 0 \rightarrow n \) do
2: \( \text{if } i \text{ is best so far then} \)
3: \( \text{pick secretary } i \text{ with probability } \frac{ix_i}{1-\sum_{j<i} x_j} \)

Since the probability that we have not chosen any previous secretary is \( (1 - \sum_{j<i} x_j) \) and the probability that this is the best secretary so far is \( \frac{1}{i} \), this algorithm would accept the secretary at \( i \) with probability \( x_i \). A formal proof can be given by induction and we skip that here.

One can exhibit a solution to this LP that has objective value \( 1/e \). Moreover, one can show by LP duality that \( 1/e \) is the best possible, and hence by the above connections, so secretary strategy can perform better.

### 2.2 The Incentive Comparable Version as an LP

We can solve the incentive comparable secretary problem in a similar way as we solved the classical version. Let \( p \) be the probability (over all possible \( n! \) orderings) that we will hire a secretary at position \( i \). Note that \( p \) does not depend on \( i \) by the incentive compatibility assumption.

Also, we note here that unlike above, at step \( i \), the algorithm might hire secretary \( i \) even if it is not the best so far to satisfy the incentive compatibility condition. We construct the following LP:

Let \( f_i = \Pr(\text{selecting the secretary at position } i \mid i \text{ is the best candidate}) \)

\[
\begin{align*}
\text{Maximize:} & \quad \sum_i \frac{1}{n} f_i \\
\forall i, & \quad f_i \leq ip \\
\forall i, & \quad f_i + (i - 1)p \leq 1
\end{align*}
\]

First we will show, like before, that any algorithm for the incentive comparable secretary problem will be a feasible solution to this linear program. For any algorithm, since \( p \) is the probability
of choosing the $i^{th}$ secretary, it can not be more that the probability of choosing the secretary at position $i$ given that it is the best so far ($f_i$) times the probability that it is the best so far ($\frac{1}{i}$). Thus the probabilities for $f_i$ taken from any algorithm will satisfy the first condition. The second condition of the linear program is simply a sum of two mutually exclusive probabilities (selecting the secretary at position $i$ or selecting one of the previous secretaries) and insisting that the probability is no more than 1, which is clearly true.

To see the objective function, conditioned on secretary $i$ being the best so far (which has exactly $1/i$ chance), there is exactly $i/n$ chance that it is the best overall. As the algorithm at step $i$, accepts the best so far, with probability $f_i$, the expected value of picking best secretary at position $i$ is $f_i \cdot (1/i) \cdot (i/n) = f_i/n$. Thus any algorithm for the incentive comparable secretary problem will imply a solution to the linear program.

As before, one can also show that any LP solution gives an incentive compatible strategy with the same value. From here one can solve the linear program to obtain a solution with value $1 - 1/\sqrt{2}$. By duality one can also show that no algorithm can do better. We refer to the BJS paper for the calculations.

3 Non-Adaptive Boxes With Linear Programing

We will now use a similar technique for the box problem. You know the distribution on the boxes and we can open any $k$ of them in any order. We will give a non-adaptive strategy that is $O(1)$-competitive to the best adaptive strategy. To do this we need a lower bound on the best adaptive policy. The difficulty is that adaptive policies are really complex trees with many paths from root to leaf. To have a simpler model to work with we will show that any such adaptive policy will correspond to a feasible solution to the following linear program. In doing this, we can use the solution to the linear program as a relaxation for all adaptive policies. As usual the probabilities below are over all the possible actions of the adaptive strategy.

Let $y_i$ be the probability that box $i$ is opened. Let $z_{jv}$ be the probability that box $j$ selected given that the observed reward is $v$. Let $f_{jv}$ be the probability that reward of box $j$ is $v$.

Maximize: $\sum_{j,v} vz_{jv}$

$\forall i, \sum_i y_i \leq k$ \hspace{1cm} (You can only open $k$ boxes)

$\sum_{j,v} z_{jv} \leq 1$ \hspace{1cm} (You can only select 1 box)

$\forall i, \ y_i f_{iw} \geq z_{iv}$ \hspace{1cm} (The Probability Chain Rule)

Note that any adaptive strategy for the box problem will give a feasible solution to this LP, thus if we can get a strategy that is competitive with the optimum solution for this LP it must be competitive with the optimal adaptive strategy. We construct our strategy as follows:

Note that we are forced to stop when we reach $k$ boxes, and the way we have designed this algorithm we do not choose the best value that we have seen or even wait to open $k$ boxes. All of this makes our job harder, but surprisingly we can still do very well.
Algorithm 2 Algorithm

1: Solve the above LP.
2: for \( j = 1 \rightarrow n \) do
3:  Open box \( j \) with probability \( \frac{y_j}{4} \).
4:  Select the box with probability \( \frac{z_{jv}}{y_jf_{jv}} \) upon seeing reward \( v \).
5:  if opened \( k \) boxes then
6:    break

Note that by construction:

\[
P(\text{Box } j \text{ is selected with value } v | \text{ Algorithm reaches } j) \\
= \frac{z_{jv}}{y_jf_{jv}} P(\text{Box } j \text{ is opened and its value is } v | \text{ Algorithm reaches } j) \\
= \frac{z_{jv}}{y_jf_{jv}} P(\text{Box } j \text{ is opened} | \text{Algorithm reaches } j) \\
= \frac{z_{jv}y_j}{y_j} = \frac{z_{jv}}{4}
\]

Note that the chance that \( k \) boxes have been opened before box \( j \) is never more than \( \frac{1}{4} \) by Markov’s inequality. Also, note that the chance that someone has been selected before box \( j \) is \( \sum_{i<j,v'} \frac{y_i}{4} \frac{z_{iv'}}{y_if_{iv'}} f_{iv'} \). Combining these facts we get:

\[
P(j \text{ is selected with value } v) \geq \frac{z_{jv}}{4} (1 - P(\text{someone was selected earlier}) - P(k \text{ boxes were opened up earlier})) \\
\geq \frac{z_{jv}}{4} \left( \frac{3}{4} - \sum_{i<j,v'} \frac{z_{iv'}}{4} \right) \geq \frac{z_{jv}}{8}
\]

Finally, we can calculate a lower bound on the reward by taking the weighted sum over all possible boxes and rewards:

\[
\sum_{j,v} P(j \text{ is selected with value } v)v \geq \sum_{j,v} \frac{z_{jv}v}{8} = \frac{\text{opt}}{8}
\]

Thus with this method we get \( \frac{1}{8} \) of the LP solution. More work on this problem and tighter bounds can be found in Dean et al. [1]

References
