The Odds Theorem

E8100/B9801 Class Project

1. Outline

This project is on the odds algorithm and its applications. I came across the odds algorithm almost by mistake, as I was researching the secretary problem, and was so struck by its elegance and ability to deal with many variations of the secretary problem that I decided to base my project on it.

I will begin by introducing the odds algorithm, and applying it to the classical secretary problem. I will then cover a number of applications of the odds algorithm: group interviews, continuous-time problems, problems with rejection and problems in which the secretary values are IID. I will end by discussing some other literature that I did not cover in this project.

Before we begin, we give a quick description of the secretary problem for those unfamiliar with it. The classical secretary problem has appeared under a wide variety of names (the dating problem, the dowry problem, the googol game, etc…) and was first published in Martin Gardner's *Mathematical Games* column in the February 1960 issue of *Scientific American*. It has since then been solved an extended a large number of times, and has evolved into what can almost be considered a field of its own (see Ferguson (1989) for an informative review of the history of the secretary problem). The setting is as follows: an interviewer needs to hire a single secretary, and sets out to interview a fixed number of candidates. While interviewing a candidate, the interviewer ascertains how the candidate ranks compared to every previous candidate. After each candidate is seen, the interviewer can either accept the candidate, and end the interview process, or reject the candidate, without any chance of ever returning to that candidate.

The secretary problem seeks the best strategy to adopt in this case. Clearly, choosing an early candidate is a bad idea – indeed, having seen very few candidates, it is difficult to know what is available. Similarly, waiting till the last candidate might also not be the best choice – the last candidate could be lousy! Formally, the secretary problem seeks a strategy that maximizes the probability of picking the *best* candidate out of all those available.

2. The Odds Algorithm

In this section, we introduce the *odds algorithm*, developed by Bruss (2000), as an alternative and more general framework to analyze stopping problems (of which the Secretary Problem is a special kind). We will discuss the ways in which this odds algorithm applies to the secretary problem in the next section.

The odds algorithm was designed to deal with problems of the following kind. Consider a fixed number (say n) of indicators $\mathbb{1}_1, \cdots, \mathbb{1}_n$. The value of each indicator (success or failure) is revealed successively. Our aim is to stop at the *last* successful indicator. In other words, if we reach the last indicator without having stopped at a success, we lose. Similarly, if we stop at a success but another success arises later, we also lose.

It is helpful, before diving into theoretical details, to consider a simple example that illustrates the essence of the way the odds algorithm deals with this problem.

> ***Example***: Consider a game that consists of throwing a fair, six-sided die n times, and whose aim is to stop at the *last* 6 obtained. (This is clearly a problem of the form above – indicator k is simply $\mathbb{1}_k = \mathbb{1}_{\left\{k^{\text{th}} \text{ throw is a 6}\right\}}$, and we would like to stop at the last successful indicator).
>
> Imagine that we had an oracle capable, at any point in the game, of answering the question "in my remaining throws, will I throw exactly one 6?". The problem would then be simple – we would ask the oracle this question after each throw, and the moment the

oracle tells us we only have one 6 left, we simply stop at that 6 whenever it appears.

Unfortunately, no such oracle is available. However, what we can do is ask "in my remaining throws, what is the *probability* that I only have one 6 left?" The answer to that question is (using simple properties of the Binomial distribution)

$$\mathbb{P}\Big(\text{Obtaining } \textit{one} \text{ 6 in last } \ell \text{ throws}\Big) = {}^{\ell}C_1 \, \tfrac{1}{6} \cdot \left(\tfrac{5}{6}\right)^{\ell} = \ell \, \tfrac{1}{6} \cdot \left(\tfrac{5}{6}\right)^{\ell}$$

Differentiating this expression and setting it to 0, we find it is maximized at $\ell = 6$ (or $\ell = 5$).

Intuitively, the most sensible strategy therefore seems to wait till we only have six throws left (at which point we maximize the probability of only one 6 left to throw), and then choose the first 6 that occurs after that. Clearly,

$$\begin{aligned}
&\mathbb{P}\Big(\text{Strategy leads to the last 6}\Big) \\
&\qquad = \mathbb{P}\Big(\text{Obtaining } \textit{one} \text{ 6 in last 6 throws}\Big) \\
&\qquad = \left(\tfrac{5}{6}\right)^{6} = 0.3349
\end{aligned}$$

□

In a nutshell, the Odds Algorithm takes exactly this approach – waiting till the point at which the probability of obtaining a *single* successful indicator is maximized – but in a more general setting. We now state and prove the odds theorem and the odds algorithm.

Theorem (The Odds Theorem): Let $\mathbb{1}_1, \cdots, \mathbb{1}_n$ be a sequence of independent indicators, with $p_j = \mathbb{E}(\mathbb{1}_j) = 1 - q_j$. Further let $r_j = p_j / q_j$. Suppose further that we observe the indicators sequentially and that our aim is to stop at the last indicator equal to 1 (at the last "success").

Then there exists an optimal rule which maximizes the probability of stopping at the last success, and it prescribes to stop on the first index $k \geq s$ such that $I_k = 1$ (if any), where s is given by

$$s = \max\left\{1, \sup_k \left\{k \in \left\{1,\cdots,n\right\} : \sum\nolimits_{j=k}^n r_j \geq 1\right\}\right\}$$

The probability that this strategy will stop at the last success is

$$V(n) = \left(\prod\nolimits_{j=s}^n q_j\right)\left(\sum\nolimits_{j=s}^n r_j\right)$$

Proof: Our proof follows Bruss (2000), with modifications. Assume $p_j < 1$ for all j. Let $S_k = \mathbb{1}_k + \cdots + \mathbb{1}_k$.

- ***Step 1***: Let $g_i(\theta)$ and $G_k(\theta)$ be the probability generating functions of I_i and S_k respectively. From independence of the I_j, we have

  $$g_t(\theta) = q_j + p_j\theta \qquad G_k(\theta) = \prod\nolimits_{j=k}^n q_j\left(1 + r_j\theta\right)$$

  Differentiating log G_k with respect to θ, and remembering that $p_j < 1\ \forall j \Rightarrow G_k(0) > 0$, we find that

  $$\log\left[G_k(\theta)\right]' = \frac{G_k'(\theta)}{G_k(\theta)} = \sum\nolimits_{k=k}^n \frac{r_j}{1 + r_j\theta}$$

  Using the properties of probability generating functions, we therefore find that

  $$\begin{aligned}\mathbb{P}\left(S_k = 1\right) &= G_k'(0) \\ &= \left(\sum\nolimits_{j=k}^n r_j\right)G_k(0) \\ &= \left(\sum\nolimits_{j=k}^n r_j\right)\left(\prod\nolimits_{j=k}^n q_j\right) \\ &= V_k(n)\end{aligned}$$

- ***Step 2***: Consider a strategy that ignores all successes up to a deterministic time k, after which it chooses the next success. Clearly, such a strategy chooses the last success if and only if $S_k = 1$. As such
  - The probability of success of such a strategy is precisely $V_k(n)$, thus proving the second part of our Theorem.

- The optimal k will maximize $p^* = \mathbb{P}(S_k = 1)$. To find the optimal k, note that, from our expression for $V_k(n)$, we can deduce the following

$$
\begin{aligned}
&V_k(n) \text{ increasing in } k \\
&\qquad \Leftrightarrow V_k(n) < V_{k+1}(n) \\
&\qquad \Leftrightarrow \left(r_k + \sum_{j=k+1}^n r_j\right)\left(q_k \prod_{j=k+1}^n q_j\right) \\
&\qquad\qquad < \left(\sum_{j=k+1}^n r_j\right)\left(\prod_{j=k+1}^n q_j\right) \\
&\qquad \Leftrightarrow q_k\left(r_k + \sum_{j=k+1}^n r_j\right) < \left(\sum_{j=k+1}^n r_j\right) \\
&\qquad \Leftrightarrow q_k r_k < \left(1 - q_k\right)\sum_{j=k+1}^n r_j \\
&\qquad \Leftrightarrow q_k \tfrac{1-q_k}{q_k} < \left(1 - q_k\right)\sum_{j=k+1}^n r_j \\
&\qquad \Leftrightarrow \sum_{j=k+1}^n r_j > 1
\end{aligned}
$$

  The sum on the LHS of the last line is monotonically decreasing as k increases. As such, the function $\mathbb{P}(S_k = 1)$ increases up to a certain value of k and then decreases – it is unimodal. There is therefore a single maximum attained at the largest k for which $\sum_{j=k+1}^n r_j < 1$, or alternatively, the largest k for which $\sum_{j=k}^n r_j \geq 1$, precisely as stated in the Theorem.

  [Note: we have implicitly assumed that $r_2 + \cdots + r_n \geq 1$. If this were not the case, there would be *no* exploratory phase (this explains the "double maximum" in the statement of the Theorem). Dealing with this case is trivial – see the paper for details].

- ***Step 3***: Our last step is to show that there are no better strategies than those defined in step 2. Consider the most general kind of strategy which ignores all successes up to a *random* time W such that $\left\{W = k\right\} \in \sigma\left(\mathbb{1}_1, \cdots, \mathbb{1}_k\right)$, and then picks the first success. Every strategy can be reduced to one of this form, because W can depend on *any* past information, and it is trivial that any strategy will restrict itself to picking a success. First note that by independence

$$\mathbb{P}\left(S_W = 1 \mid \mathbb{1}_1, \cdots, \mathbb{1}_k, W = k\right) = \mathbb{P}\left(S_k = k\right)$$

We have shown, however, that $\mathbb{P}\left(S_k = k\right)$ is unimodal. As such, it will be maximized at one point only, and the optimal random variable W will have point mass at the same k as that described above. Thus, the above strategy is indeed optimal. This proves our theorem. ■

Based on this theorem, the odds algorithm is therefore rather simple

Algorithm (The Odds Algorithm): Let $\mathbb{1}_1, \cdots, \mathbb{1}_n$ be a sequence of independent indicators, with $p_j = \mathbb{E}(\mathbb{1}_j) = 1 - q_j$. Further let $r_j = p_j / q_j$.

The rule which maximizes the probability of stopping at the last success, prescribes to stop on the first index $k \geq s$ such that $I_k =$ 1 (if any), where s is given by

$$s = \max\left\{1, \sup_k \left\{k \in \left\{1, \cdots, n\right\} : \sum_{j=k}^{n} r_j \geq 1\right\}\right\}$$

The probability that this strategy will stop at the last success is

$$V(n) = \left(\prod_{j=s}^{n} q_j\right)\left(\sum_{j=s}^{n} r_j\right)$$ □

3. Application to the classical secretary problem

Recall that our classical treatment of the Secretary Problem with n secretaries led us, asymptotically, to reject the first n/e secretaries and to then accept the best secretary so far, if any. Asymptotically, we found that the algorithm picked the best secretary with probability $1/e$

(see, for example, Freeman, 1983). In this section, we briefly show that the Odds Algorithm leads to the same result.

Let $\mathbb{1}_1,\cdots,\mathbb{1}_n$ be indicators such that $\mathbb{1}_k = 1$ if secretary k is the best secretary seen so far, and $\mathbb{1}_k = 0$ otherwise. Clearly, $p_k = \mathbb{E}(\mathbb{1}_k) = 1/k$, and so $r_k = \frac{1}{k-1}$ (using the terminology above). It is fairly obvious that restated using this framework, the secretary problem simply aims to stop at the *last* successful indicator from this sequence – in other words, the last secretary that was "better than all the ones before it". Using the Odds Approach, we would stop on the first index $k > s$ such that secretary k is the best seen so far, where

$$s = \max\left\{1, \sup_k \left\{k \in \left\{1,\cdots,n\right\} : \sum_{j=k}^n \tfrac{1}{j-1} \geq 1\right\}\right\}$$

Approximating the sum as an integral, we require

$$\int_s^n \tfrac{1}{j-1}\ \mathrm{d}j \approx 1 \Rightarrow s \to n/e \text{ as } n \to \infty$$

Similarly, the Odds Algorithm predicts that the probability of choosing the best secretary is

$$\begin{aligned} V(n) &= \left(\prod_{j=s}^n \tfrac{j-1}{j}\right)\left(\sum_{j=s}^n r_j\right) \\ &= \tfrac{s-1}{n}\left(\sum_{j=s}^n r_j\right) \\ &\to \tfrac{1}{e} \text{ as } n \to \infty \end{aligned}$$

Both results are consistent with those obtained in the classical case.

4. The Secretary Problem with Group Interviews

We now consider our first non-classical application of the secretary problem, also covered by Bruss (2000) – a situation in which instead of interviewing secretaries one by one, we interview them in n groups of size $\ell_1,\cdots,\ell_n$. After each group interview, we can choose to accept one secretary from that group, or reject the entire group, without any opportunity of recall.

To use the odds algorithm, we need to re-cast our problem in terms of the indicator framework defined above. In this case, consider indicators $\mathbb{1}_1,\cdots,\mathbb{1}_n$, where $\mathbb{1}_k = 1$ if group k contains the best secretary seen so far, and 0 otherwise. In this case,

$$p_k = \mathbb{E}(\mathbb{1}_k) = \frac{\ell_k}{\sum_{i=1}^k \ell_k} = \frac{\ell_k}{b_k}$$

(To see why, consider that the probability any given candidate in group k is the best of all b_k candidates is $1/b_k$, but since there are ℓ_k candidates in that group, we get ℓ_k "chances" are trying to find that best candidate). We then have

$$r_k = \frac{p_k}{1 - p_k} = \frac{\ell_k}{b_k - \ell_k} = \frac{\ell_k}{b_{k-1}}$$

As such, the odds algorithm then simply prescribes the policy of stopping at the first item $k > s$, where

$$s = \sup\left\{k : \sum_{i=k}^{n} \frac{\ell_k}{b_{k-1}} \geq 1\right\}$$

This is precisely the result obtained by Hsiau & Yang (2000) after much painstaking analysis, in a paper exclusively devoted to studying this problem. The odds theorem allows us to obtain this solution almost trivially.

5. A continuous-time secretary problem

Our next application is also covered by Bruss (2000), and concerns the somewhat more complicated problem of a secretary problem in continuous time. More specifically, we consider a situation in which the interviewer sets an interview period from $t = 0$ to $t = T$. Within that period, candidates arrive according to an inhomogeneous Poisson process, with density $\lambda(t)$.

We analyze this problem by splitting the period $[0,\ T]$ into segments $\Delta_k = (t_{k-1}, t_k]$. Assume that we have a function, $h(t)$, defined as follows

$$h(t) = \mathbb{P}\left(\text{Secretary best so far} \mid \text{Secretary arrives at time } t\right)$$

Given this function, the probability of getting a secretary that is the "best so far" during the interval Δ_k is

$$\begin{aligned} p_k = {} & \mathbb{P}\left(\text{One secretary arrives in that interval}\right)\mathbb{P}\left(\text{That secretary is the best so far}\right) \\ & + \mathbb{P}\left(\text{More than one secretary arrives in that interval}\right) \times \text{constants} \end{aligned}$$

Using standard properties of inhomogeneous Poisson processes, we can write this as

$$p_k \approx \lambda(t_k) h(t_k) \Delta_k$$

[Note that this is equivalent to thinning the Poisson process using the function $h(t_k)$ – this seems sensible; we are only interested in arrivals that are the best so far]. Now, consider that the *odds* of a "best secretary so far" arriving in Δ_k is

$$r_k = \frac{p_k}{1-p_k} \approx \frac{\lambda(t_k)h(t_k)}{1-\lambda(t_k)h(t_k)\Delta_k}\Delta_k \sim_{\Delta_k \to 0} \lambda(t_k)h(t_k)\Delta_k$$

So we see that as our sub-intervals of time get infinitely fine, we obtain a well-defined density of the "odds of getting a best secretary so far", given by $\varphi(t) = \lambda(t)h(t)$.

We are now ready to apply the odds theorem, which would simply prescribe accept the first secretary to arrive at a time $t > s$, where s is given by

$$s = \sup\left\{t \in [0,T] : \int_t^T \lambda(u)h(u)\ \mathrm{d}u \geq 1\right\}$$

The last detail we need to consider is how to find the function $h(t)$ used above. The mechanics are not complicated, we simply proceed as follows:

$$\begin{aligned} h(t) &= \mathbb{P}\big(\text{Arrival } k \text{ a success} \mid \text{Arrives at time } t\big) \\ &= \mathbb{P}\big(\text{Arrival } k \text{ largest so far} \mid \text{Arrives at time } t\big) \\ &= \mathbb{E}\Big(\mathbb{P}\big[\text{Arrival } k \text{ largest so far} \mid m \text{ items in } [0,t)\big]\Big) \\ &= \mathbb{E}\left(\frac{1}{m+1}\right) \\ &= \mathbb{E}\left(\left[1+\mathrm{Po}\left(\int_0^t \lambda(u)\ \mathrm{d}u\right)\right]^{-1}\right) \end{aligned}$$

There is no simple closed-form expression for this last integral, but it can be calculated numerically for a grid of values of t.

6. The secretary problem with rejection

We now consider an application of the secretary problem in which, once a candidate is offered the job, they might *refuse* to accept it with probability $1 - p$ (independent of everything else). This problem, while seemingly quite simple, is actually quite subtle and presents a number of interesting issues. The first of these is that our *objective* in this case is no longer as simple as it was in the classical secretary problem. In particular, the classical objective can be extended in one of three ways:

1. ***"Best of all" objective*** – this would imply that we still seek to pick the best candidate of all, regardless of whether that best candidate would accept the offer or reject it. Though this is the most obvious extension of the classical objective, it is somewhat intuitively unpalatable, because in implies that in some regions of the sample space (namely, those in which the best candidate rejects a job offer), it is *never* possible to reach our objective.
2. ***"Best of all available" objective*** – this is the next most obvious generalization of the classical objective, in which our aim is to pick the best candidate *out of those that would accept the job*. This ensures that it is always possible to reach our objective, regardless of what part of the sample space we find ourselves in, and is intuitively appealing. (Note that with this objective, the problem is somewhat analogous to a secretary problem with a random number of candidates, binomially distributed with certain known parameters).
3. ***"No regrets" objective*** – in this framework, the interviewer's objective is to have "no regrets". That is to say, the aim of the interviewer is *never* to find himself in a situation in which his offer of employment is rejected by a candidate which leads him to have to offer the job to a subsequent, less qualified candidate. Or, put another way, the interviewer ideally wants to offer the job to the last candidate that is the best so far *and* will accept the job. I had a short discussion with Prof Bruss about this particular objective, and he agreed that in many cases, it is the most psychologically intuitive choice.

We consider each of these cases in turn

"Best of all" objective

This objective is the simplest generalization of the classical objective. It is perhaps surprising, therefore, that it is much more difficult to model using the indicator approach. Indeed, we might be tempted to use the following sequence of indicators

$$\mathbb{1}_k = \begin{cases} 1 & \text{secretary } k \text{ is the best so far } \textit{and} \text{ accepts the job} \\ 0 & \text{otherwise} \end{cases}$$

with $p_k = p/k$. Unfortunately, this method is ultimately unsuccessful. Indeed, consider that part of the sample space in which the best secretary rejects employment. There, it should be *impossible* to achieve our objective. However, it is clear that even in that part of the sample space, some of the indicators $\mathbb{1}_k$ will be successful, and that it is therefore possible to be successful in our aim of choosing the last successful indicator. Thus, it is clear that the indicator approach does not adequately model the secretary problem with this objective.

To accurately model this problem using the indicator approach, we would require a sequence of indicators in which $\mathbb{1}_k$ was equal to 1 if secretary k is the best so far, *and* accepts the job, *and* the best secretary overall accepts the job. Unfortunately, these events are not independent, which p_k difficult to calculate. More problematically, however, the resulting sequence of indicators is not independent, which means the odds theorem does not apply in this case.

Smith (1975) and Freeman (1983) provide a different argument, and they argue that the best strategy for this particular secretary problem is to pick the best secretary $k > s$, where s is asymptotically given by $s = np^{1/(1-p)}$.

We verified this result by simulation. We considered every policy with the structure above (ie: ignore all secretaries until s, and then pick the best so far) for every value of s, and plotted the resulting probability of obtaining the best secretary overall, based on 150,000 simulations. The results, plotted in Figure 1, were obtained for 500 sectaries and $p = 0.7$. Note, as we might expect, that that maximum possible probability of achieving our aim is less than the $1/e$ of the classical secretary problem, due to that part of the sample space in which it is impossible to attain our objective.

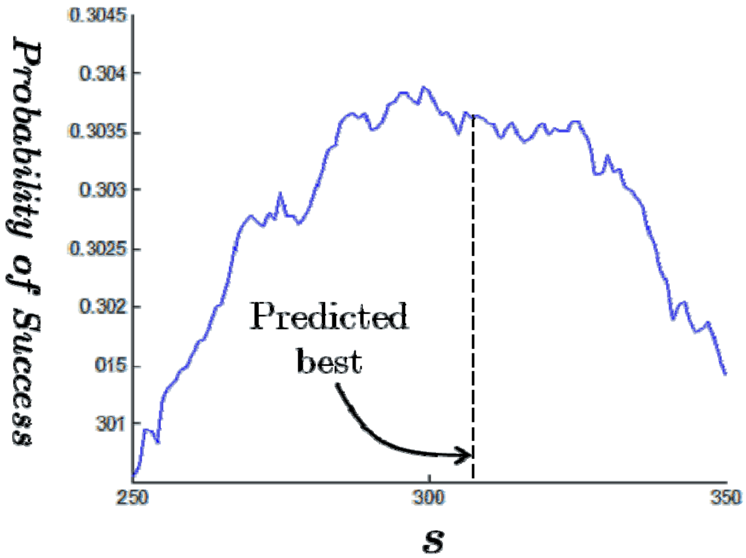

Figure 1: Simulation for the "*best of all*" objective

"No Regrets" objective

We now consider the third objective mentioned above, in which our aim is to pick the last secretary that is the best so far *and* accepts employment. In this case, the indicator approach clearly applies directly with the indicator mentioned above; namely

$$\mathbb{1}_k = \begin{cases} 1 & \text{secretary } k \text{ is the best so far } \textit{and} \text{ accepts the job} \\ 0 & \text{otherwise} \end{cases}$$

We then have $p_k = \mathbb{P}(\text{Accepts}) \cdot \mathbb{P}(\text{Best so far}) = p / k$, and

$$r_k = \frac{p}{k - p}$$

Applying the machinery developed above, we find that the optimal policy asymptotically has $s \approx ne^{-1/p}$. We verified this policy using 500 secretaries, $p = 0.7$ and 150,000 simulations, and reported our results in Figure 2. (See the comments preceding Figure 1 for a guide on how to read this figure). The odds algorithm also allows us to predict the probability of success of *any* stopping time strategy. We also plotted this curve on our graph, and found that it matched our simulations rather well.

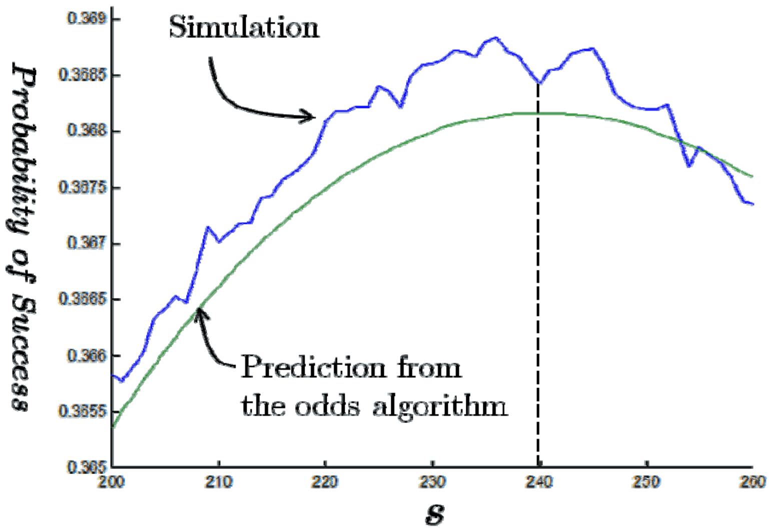

Figure 2: Simulation for the "*no regrets*" objective. The dotted line indicates the best optimal time prescribed by the odds algorithm

"Best of all available" objective

We now move on to our third, and arguably most intuitive, objective for the secretary problem with rejection, in which our aim is to pick the best candidate *out of those which accept our offer*. Again, the situation is relatively easily modeled by using indicators of the following form

$$\mathbb{1}_k = \begin{cases} 1 & \text{Candidate } k \text{ best of those accepted so far } \textit{and} \text{ accepts employment} \\ 0 & \text{otherwise} \end{cases}$$

A quick glance at this equation, however, quickly reveals some additional complications. In particular, it is unclear how the interview discovers whether a candidate will accept or reject employment. There are two possibilities:

- ***Full-information setting***: The interview finds out whether the candidate would accept or reject employment during the course of the interview, before an offer is potentially made.
- ***Partial-information setting***: The interviewer only finds out whether a candidate would or would not accept employment if he makes an offer to the candidate, and the candidate refuses/accepts it.

The second setting above is much more complicated, and not so readily modeled using the indicator approach. See Tamaki (1991) for a treatment of the problem in that setting.

We do, however, consider the first setting in more detail. The difficulty arises, even in this simple case, with the need to calculate $p_k = \mathbb{E}(\mathbb{1}_k)$. My original, naïve approach was to simply condition on the number of candidates who had accepted so far (clearly binomially distributed, with probability p), and to proceed as follows:

$$\begin{aligned} p_k &= \mathbb{E}\left(\mathbb{1}_{\{\text{Secretary } k \text{ best of those accepted so far } \textit{and} \text{ accepts employment}\}}\right) \\ &= p\mathbb{E}\left[\mathbb{E}\left(\mathbb{1}_{\{\text{Secretary } k \text{ best of those accepted so far}\}} \mid m \text{ accepted so far}\right)\right] \\ &= p\mathbb{E}\left(\frac{1}{m+1}\right) \\ &= p\mathbb{E}\left(\frac{1}{1+\mathrm{Bin}[k-1,p]}\right) \end{aligned}$$

I then applied the classical odds theorem, as described above. (Note that the inverse moments of the binomial distribution had to be obtained numerically – no closed-form analytic expression is available for them. In practice, there exist some power series expansions for these moments – see, for example, Rempala (2004). The results of this method are illustrated in Figure 3. Once again, the odds algorithm correctly predicts the probability of success for various constant-stopping-time rules.

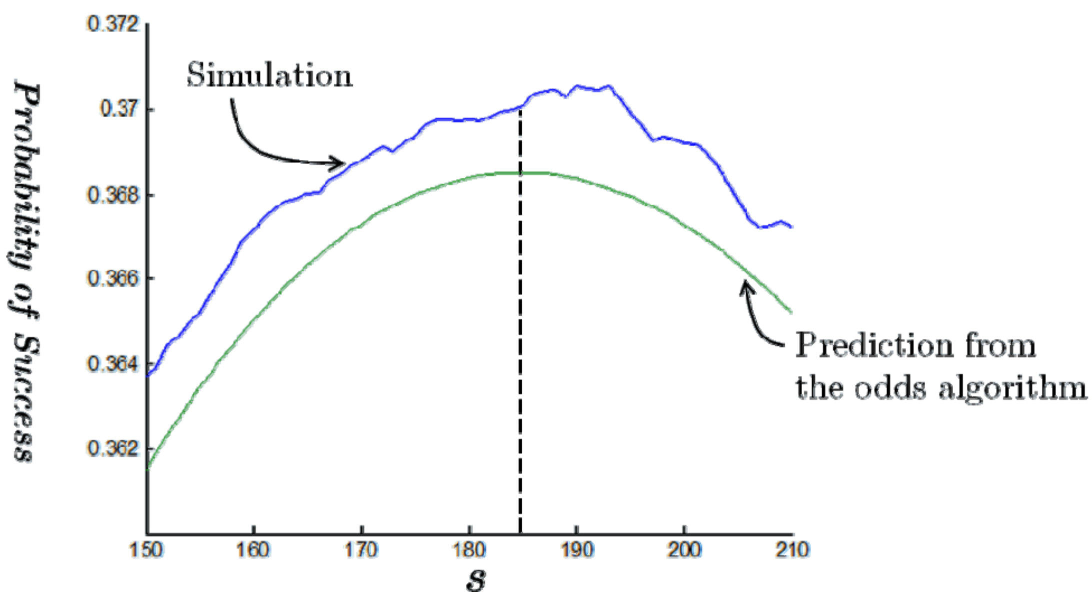

Figure 3: Naïve simulation for the "*best of all available*" objective

After some additional thought, however, I realized that the algorithm above was in fact *too* naïve, and that the conditions of the odds theorem were in fact not satisfied in this particular situation. Indeed, even though the indicators in this case are still independent from each other, *additional* information becomes known to the interviewer as the interview progresses, which has a bearing on the value of these indicators. Indeed – at the start of the interview process, the interviewer can only guess as to how many candidates will accept or reject offers of employment. As the interviews progress, this number becomes less and less uncertain.

I found out (unfortunately too late!) that this generalization of the odds theorem (in which additional information is revealed as the indicators are seen) was actually dealt with by Ferguson (2008). In this project, I chose instead to take a more pedestrian approach. Instead of calculating *one* stopping time s before all interviews (as we did in the classical secretary problem), I decided to constantly re-calculate the time s after every candidate was interviewed, *based on information that had been elicited thus far*. More precisely, my heuristic algorithm proceeded as follows:

> ***Algorithm (Heuristic for the Odds Algorithm with information elicitation)***: Let $\mathbb{1}_1, \cdots, \mathbb{1}_n$ be a sequence of independent indicators.
>
> Begin observing indicators. Whenever a successful indicator is observed, re-calculate the stopping time s using the method in the classical odds algorithm, *conditional on all the data observed so far*. If, at any time, the calculated stopping time becomes smaller than or equal to the index of the successful indicator that was just observed, stop at that indicator.

The difficulty, in this case, then arose from the need to derive $p_k = \mathbb{E}(\mathbb{1}_k)$ *conditional on past information*. We proceeded as follows. Suppose we have thus far observed v candidates, and that x of these have been accepted (including v if applicable). Then

$$
\begin{aligned}
p_k &= \mathbb{E}\left(\mathbb{1}_{\{\text{Best of those accepted so far } \textit{and} \text{ accepts employment}\}}\right) \\
&= p\mathbb{E}\left(\mathbb{E}\left[\mathbb{1}_{\{\text{Best of those accepted so far}\}} \mid m \text{ accepted so far}\right]\right) \\
&= p\mathbb{E}\left(\frac{1}{m+1}\right) \\
&= p\mathbb{E}\left(\frac{1}{x+\mathrm{Bin}[k-v-1,p]+1}\right)
\end{aligned}
$$

Applying this modified algorithm did provide some marked improvements over the naïve algorithm. Figure 4 graphs these improvements, which are clearly more dramatic for smaller number of secretaries.

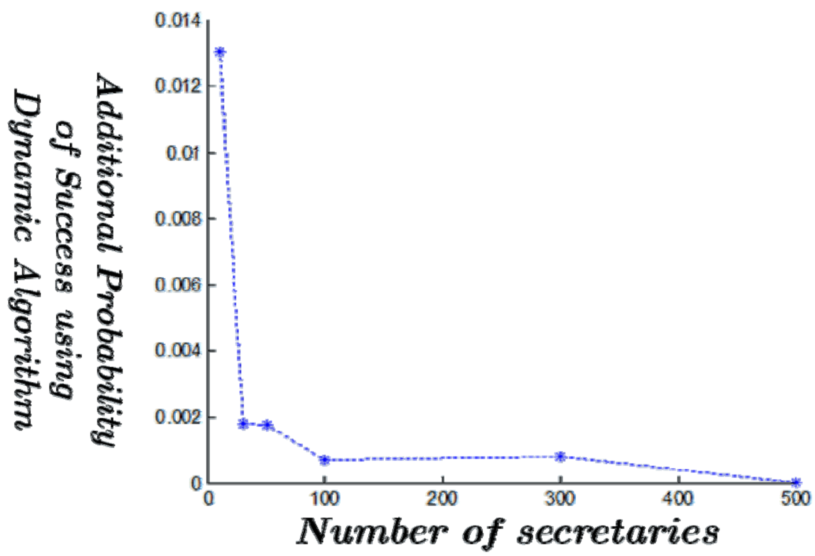

Figure 4: Improvements obtained by using a dynamic algorithm instead of the naïve static algorithm for the "*best of all available*" objective in the secretary problem

7. Extension to the IID Case

Our last example will be an extension of the classical secretary problem in which the values of secretaries are independently and identically distributed. Our objective is still to pick the best secretary of all, but we now have some more information as to the structure of secretary values.

There is some literature on this problem. Gilbert and Mosteller (1966) solve a version of this problem in which the distribution of secretary values is fully known, and Tamaki (2009, 2010) solves the same problem using an extension to the odds approach discussed in this paper.

In this report, however, we will once again use the more pedestrian modified odds algorithm described in the previous section, in which we recalculate the stopping threshold s each time new information is obtained. In particular, each time a secretary that is the best so far is observed, we recalculate the threshold, and accept that secretary if the threshold is less than or equal to the index of that secretary. We will only consider the simplest, full-information case, in which the distribution of secretary values is fully known. We can assume, without loss of generality, that the distribution is uniform in the interval [0,1] – all other distributions can be mapped to this distribution using the inverse-CDF method.

Before we begin developing the details of this approach, we prove the following useful lemma

> ***Lemma***: Let $X_1, \cdots, X_n$ be n independent random variables, all with a uniform distribution in the interval [0,1], and let $M = \max_{i=1,\cdots,n} X_i$. Then M has a $\text{Beta}(n,1)$ distribution.
>
> ***Proof***: Consider that for the maximum to be somewhere in the interval $[u,\ u + \mathrm{d}u]$, we require the maximum X_i to be in the range $[u,\ u + \mathrm{d}u]$, and every other X_i to be less than u. Furthermore, the are ${}^nC_1$ ways of choosing the maximum random variable. As such
>
> $$\begin{aligned}\mathbb{P}\left(M \in [u, u + \mathrm{d}u]\right) & \\ &= {}^nC_1 \mathbb{P}\left(n - 1 \text{ RVs} \le u\right) \mathbb{P}\left(\text{One RV} \in [u, u + \mathrm{d}u]\right) \\ &= nF(u)^{n-1} f(u)\mathrm{d}u \\ &= nu^{n-1}\mathrm{d}u \\ &= \frac{\Gamma(n+1)}{\Gamma(n)\Gamma(1)} u^{n-1}\ \mathrm{d}u\end{aligned}$$
>
> This is indeed the PDF of a Beta(n, 1) distribution. ■

Let us now develop this approach more fully. As before, the series of indicators we will be using are as follows

$$\mathbb{1}_k = \begin{cases} 1 & \text{Secretary } k \text{ is the best so far} \\ 0 & \text{Otherwise} \end{cases}$$

We will now derive an expression for $p_k = \mathbb{E}(\mathbb{1}_k)$ based on past observations. Imagine we have observed all candidates up to candidate v, and that we have found the maximum so far to be m, and denote the "value" of the k^{th} candidate as V_k. Then

$$\begin{aligned} p_k &= \mathbb{P}(V_k = \max_{i=1,\cdots,k}[V_i] \mid \max[V_1,\cdots,V_v] = m) \\ &= \mathbb{P}(\max[V_{v+1},\cdots,V_k] \geq m \text{ and } \max[V_{v+1},\cdots,V_k] = V_k) \\ &= \mathbb{P}(\max[V_{v+1},\cdots,V_k] = V_k)\mathbb{P}(\max[V_{v+1},\cdots,V_k] \geq m) \\ &= \frac{1}{k-v}\mathbb{P}\left(\text{Beta}[k-v,1] \geq m\right) \end{aligned}$$

Using our dynamic odds algorithm on a 500-secretary problem and using 150,000 trials, the best secretary is chosen with probability 0.5409. This compares favorably with the optimal probability of 0.5802, derived by Gilbert & Mosteller (1966) and Tamaki (2009, 2010).

8. Other References of Interest

We have barely scratched the surface of the various applications of the odds theorem and the odds algorithm. In particular, our treatment of the secretary problem with rejection, and of the secretary problem in the IDD case, were only heuristic. The papers listed in the references below contain many other applications of the concepts discussed in this paper. In particular, Tamaki's 2010 paper extends the odds theorem to a situation in which one wants to stop at the z^{th} indicator, instead of the last one, which has applications to the k-secretary problem. There is also a significant gap in the literature around the issue of the IID secretary problem in which the distribution involved is *not* known (or perhaps known up to some parametric family). One would hope that Ferguon's more general odds theorem could be used to good effect to develop an approach to dealing with this problem.

9. Acknowledgements

I should like to thank Prof Omar Besbes and Prof Vineet Goyal for introducing me to the secretary problem, and to many other exciting topics in dynamic learning, in their Spring 2011 course at the Columbia Business School. I am also very grateful to Prof Bruss who was so

responsive to emails I sent him during the course of writing this project and who provided invaluable insights into some of the topics I was looking into.

10.References

- [**Bruss, 2000**] F.T. Bruss. *Sum the Odds to One and Stop*. The Anals of Probability, 28(3):1384-1391, 2000.
- [**Ferguson, 2008**] T. S. Ferguson. *The Sum-the-Odds Theorem with Application to a Stopping Game of Sakaguchi*. Unpublished. Retrieved from www.math.ucla.edu/~tom, 29th May 2011.
- [**Ferguson, 1989**] T. S. Ferguson. *Who Solved the Secretary Problem?*. Statistical Science, 4(3):282-289, August 1989.
- [**Freeman, 1983**] P.R. Freeman. *The Secretary Problem and Its Extensions: A Review*. International Statistical Review, 51(2):189-206, 1983.
- [**Gilbert and Mosteller, 1966**] J. P. Gilbert and F. Mosteller. *Recognizing the Maximum of a Sequence*. Journal of the American Statistical Association, 61:35-73, 1966.
- [**Hsiau and Yang, 2000**] S-R Hsiau and J-R Yang. *A Natural Variation of the Standard Secretary Problem*. Statistica Sinica, 10:639-646, 2000.
- [**Rempala, 2004**] G. A. Rempala. *Asymptotic Factorial Power Expansions for Binomial and Negative Binomial Reciprocals*. Proceedings of the American Mathematical Society, 132(1):261-272, January 2004.
- [**Smith, 1975**] M.H. Smith. *A Secretary Problem with Uncertain Employment*. Journal of Applied Probability, 12(3):620-624, September 1975.
- [**Tamaki, 1991**] M. Tamaki. *A Secretary Problem with Uncertain Employment and Best Choice of Available Candidates*. Operations Research, 39(2):274-284, March 1991.
- [**Tamaki, 2009**] M. Tamaki. *Optimal Choice of the Best Available Applicant in Full-Information Models*. Journal of Applied Probability, 46:1086-1099, 2009.
- [**Tamaki, 2010**] M. Tamaki. *Sum the Multiplicative Odds to One and Stop*. Journal of Applied Probability, 47:761-777, 2010.