Online Auction and Secretary Problem

Greg Harrell, Josh Harrison, Guifen Mao, and Jin Wang
Department of Mathematics and Computer Science
Valdosta State University, Valdosta, GA 31698, USA

Abstract - In this study, we are focused on a set of problems of a very specific and popular topic; The Online Auction, Secretary Problem, and K-Secretary Problem. Detailed discussion of methods is used to obtain each problems optimal solution. Also, models to help guide the explanation. We describe the relevance of each and how they relate to one another in finding probability as well as the optimal solution. Last but not least, research in the algorithms associated in the method was discussed for finding the optimal solution.

Keywords: Secretary Problem; Online Auction; K-Secretary Problem

1. Introduction

You are in need of a home and you are trying to find the perfect environment to live in. You are given a set of locations to choose from. They are revealed to you one by one and you must make a decision to take that current location or pass on it for the remaining choices, without going back. Think about it. How would you handle this situation to have the highest probability in acquiring the best place to live in and not end up with a location that is not ideal? Choices like this and the solution, is what will be discussed in detail.

2. Secretary Problem

The Secretary Problem is a very popular problem that has great use. It aroused from trying to decide the best secretary out of a group of individuals. However, it is not just as simple as a quick selection. The challenge arrives when you have no knowledge of them, until you interview each individual one by one, each having the same probably of being selected, and you must decide right then rather to accept them or to deny with no going back.

2.1 History

This problem first appeared in the late 1950’s and early 1960’s. It is a problem that spread throughout. It was similar to the current problems at the time, for example the marriage problem. However, it had a shocking solution. With the amount of attention and growth this problem was getting, it became a field of study. In research papers, Statisticians Lindley (1961), Dynkin (1963), Moritguti, Chow, Samuels and Robbins (1964), and Gilbert and Mosteller (1966), were trying to solve this problem. They were at a race to see who would be the first to solve the problem (Thomas page 282).

2.2 Optimal Solution

The primary goal is to find the best contestants for the secretary job position. This is what you call, the optimal solution. In other terms, the optimal stopping is \( r \). To acquire such solution, a method or algorithm is necessary. For this particular problem, the strategy in finding the optimal \( r \) is to use the stop method, or the stopping rule. Selecting a set of \( r - 1 \) contestants, set \( S \), of \( n \) individuals and deny them the position automatically. Then you compare \( r \) contestants to the set \( S \). If the current contestant is superior then set \( S \), you select that contestant as the solution. If it was not superior, then you move on to the next one in line, selecting the \( n \)th contestant if no other options were superior. Varying the size of set \( S \) will give you multiple results, vary the probability of succeeding in finding the best contestant. Article [5] provides the following algorithm that will provide the probability of success, given \( r \):

\[
P(r) = \sum_{n=1}^{\infty} n (r-1)/(n-1) - \frac{1}{n} = \frac{r-1}{n} \sum_{r=1}^{n} \frac{1}{r-1}.
\]

Article [1] states, “Lindley [1961] and Dynkin [1963] proved that a generalization of this strategy to a setting with \( n \) applicants yields a probability approaching \( 1/e \approx .37 \) of hiring the best secretary, and that this is the best possible guarantee.” There has been results that help prove this statement. In article [5], it also states, “Letting \( n \) tend to infinity, writing \( x \) as the limit of \( r/n \), using \( t \) for \( i/n \) and \( dt \) for \( 1/n \)”, which can be represented by the following integral:

\[
P(x) = x \int_{x}^{1} 1/t \ dt = -\ln(x).
\]

The derivative of this integral in respect to \( X \), will prove the optimal \( x \) is equal to \( 1/e \), when solving for \( x \) and setting it equal to 0.
2.3 Secretary Model

A model was constructed to simulate the Secretary Problem using Java, a programming language. In the model, algorithm (1) was used on a set of 10 individuals, ranking them from 0-9 and rank 9 being the optimal solution. The model tested each value of $r$ to use for the stopping rule, 2-7, 1 million times and the Figure 1 resembles the average results:

<table>
<thead>
<tr>
<th>$r$</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model: $P(r)$</td>
<td>0.2827</td>
<td>0.36639</td>
<td>0.39848</td>
<td>0.39843</td>
<td>0.37249</td>
<td>0.32633</td>
</tr>
<tr>
<td>Algorithm: $P(r)$</td>
<td>0.2828</td>
<td>0.36579</td>
<td>0.39869</td>
<td>0.39823</td>
<td>0.37281</td>
<td>0.32738</td>
</tr>
</tbody>
</table>

Figure 1

The results from the model, compared with the formula, are almost identical. As the trials increase, the Model result will tend to the result from the formula, algorithm (1).

3. K-Choice Secretary Problem

The K-Choice Secretary Problem is similar to the Secretary Problem, note the name. However, there is a big difference between the two. Previously, you experienced the Secretary Problem and also seen the results of the simulation. You were introduced to a method that could be used to select the best choice of a set of n items, algorithm (1). Now, what if you want a set of k elements that are the best? This is where the K-Choice Secretary Problem was originated.

3.1 Optimal Solution

There is a process in finding the optimal solution for this problem. It is similar to the Secretary Problem, however, there is a few more steps you have to do in this situation. Article [4] states the following algorithm:

(a) Observe the first $[n/e]$ elements. (3)

(b) Remember the best k elements among these first $[n/e]$, and call this set T. If $k > [n/e]$, then let T consist of the first $[n/e]$ elements observed, together with $k - [n/e]$ "dummy elements" of zero value.

(c) Whenever an element arrives whose value is greater than the minimum-value element in T, select this element and delete the minimum-value element from T.

3.2 Theory

The theory behind the previous solution, algorithm (3), is a very ideal and optimal approach to the problem. Taking the best k elements of the set of elements that were observed based on the stopping rule, algorithm (1), and comparing them with the following elements, will result in the highest probability of obtaining the optimal solution. Theoretically this seems correct, analyzing the first set of elements to get an idea of value of possible options and comparing the rest of the elements with the top k analyzed elements. As long as you follow the stopping rule, it will appear optimal. With further study, discussed in article [4], it was discovered that the constant $e$ becomes less optimal as $k$ approaches infinity.

3.3 Example

Branching off from the previous example discussed, you want to hire a team of the top two candidates. Ten people applied for the team. Just like previously, you are unaware of the value of each candidate, until you interview them one by one. During the interview, you have to make a decision to hire them or to let them go for good. The optimal way of solving this problem is to first use the stopping rule, algorithm (1), to determine the cut off for the set that will be analyzed. Then, you form a subset, team, of $k$ candidates with the best valued options. In this example, $k = 2$ and the cut off will be 3. The team will consist of the top 2 of the set of 3 that were analyzed, set T. Once you have set T, compare each of the remaining candidates to the team. If the candidate has a greater value than the minimum member of the team, you select that candidate and remove the minimum member from the team. You continue this method until that you have selected 2 candidates or until $k$ – selected element(s) are left.

The following figure is a visual of the example:

<table>
<thead>
<tr>
<th>Ten candidates in random order, ranked from 1-10 in value</th>
<th>4, 10, 5, 3, 9, 7, 6, 2, 8, 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut off set based on stopping rule, $r = 4.$</td>
<td>4, 10, 5</td>
</tr>
<tr>
<td>Subset, set T</td>
<td>5, 10</td>
</tr>
<tr>
<td>End result after comparing the remaining candidates</td>
<td>9, 1</td>
</tr>
</tbody>
</table>

Figure 2
As you can see, in this example, the optimal team was not selected due to probability, the 10 was in the cut off set. After selecting 9 because the value was greater than 5, no remaining candidate had greater value then the remaining candidate with the rank of 10 in set T. So, the last k-selected remaining element(s) was selected along with the 9. This is the result in 9, 1.

4. Online Auction

4.1 History

Online Auction is the best source for an example of modern markets. It ranges from all different categories, more specifically, networked markets. It is a method used to sale or purchase goods on a joined network. This gives business an option in being in a more preferred environment for the buyers or even the sellers. Also, being on the web can give you more access as far as buyers, or even goods to browse through. As a seller, using the optimal stopping rule, theory, can be a great power in the online auction environment as well. Article [2] states, “This first has been done by Hajiaghayi et al. [2004] who considered the well-known secretary problem in online settings.”

4.2 Relations with K-Choice Secretary

Online Auction and the K-Choice Secretary Problem have many similarities. This is why the K-Choice Secretary Problem is a very powerful tool in the online setting. The same optimal logic that is involved with finding the best way to handle the K-Choice Secretary Problem can be used for online auction. In setting of an online auction, as a seller, you are trying to bid off product that you own at the greatest sale price possible. However, you do not want to turn down a bid that may be the highest you will ever get with the current auction. As you can see, this is a similar scenario as the K-Choice Secretary Problem. When you are having an auction online, buyers will randomly approach and bid on the current item. It is unknown how much the next person will bid, or even if there will be anyone else. So, a decision has to be made on the spot to take the current bid or not, just like the Secretary Problem or the K-Choice Secretary Problem, if it is an auction of multiple items.

5. Apply K-Choice With Online Auction

This section a Model will be built to apply the K-Choice Secretary Problem with the online setting of an auction to show that the secretary type problems can be used for an optimal strategy for Online Auction environment.

5.1 Application Model

The model simulation is based on an auction of cars. There will be a total of 100 cars that are up for auction that individuals can bid on. For testing purposes, there will be a set of 300 buyers that will be randomly selected to bid one by one. This process will continue until there are no more cars.

5.2 Optimal Strategy

To find the optimal solution to this application model, the same strategy will be used as the one used for the K-Choice Secretary Problem. Review algorithm (3) for details. Once a buyer is found that has the bid amount needed based on the analysis, we will accept that buyer and sell one car from auction.

5.3 Algorithm: Pseudo Computer Code

Int k = 100; Int n = 300; Int r = 111, based on cut off rule 1/e;
Array[] buyers = Group of random ordered buyers with one bid value;
Array[] setT = Top k buyers from the 110 (r – 1) analyzed set based on cut off rule, r;
Array[] acceptedBuyers = The buyers that are accepted based on comparison of first analyzed set of buyers.
While( k != 0) {
    Current = the next available buyer that appears with their bid.
    If( number of left over buyers is equal to number of left over cars left for auction)
        Break, and default accept all the left over buyers.
    If( Current’s bid > then minimum buyer’s bid in set T)
        Remove minimum buyer from set T;
        Add Current to acceptedBuyers set.
        Sell the car the current buyer, k – 1;
}
Return acceptedBuyers;
//No more cars to auction

6. Extended Research

Recently there was discussion about the optimal solution for the secretary problem and the corresponding optimal solution. There were two algorithms, one for small amount of n elements, algorithm (1), and another that shows as n attends infinity, the cut off rule is 1/e, algorithm (2). I decided to research and use a model to see what happens to the optimal r from algorithm (2) as n tends to infinity.

6.1 Model

The model to obtain information on what happens to the accuracy of the optimal r produced by algorithm (2) was designed to use both algorithms and analyze both
results. There will be multiple tests on multiple n sizes. The model will produce the portion of P(r) results from algorithm (1) around the optimal r produced by the algorithm (2), limiting excess data.

### 6.2 Results Discussion

The results of the previous model can be found on next page. There were four tests produced using this model. Examining each one, you can see the lower the n value, the smaller the variance. Using $1/e$ to determine the cut off rule seemed to be accurate, when rounded, roughly below 100. With further testing, you can see the variance increase from the optimal r produced by both algorithms. Testing the value $n = 24,333$, algorithm (2) produced $r = 9003.210$. However, based on the results from algorithm (1), the optimal r was found $\approx 8934 – 8970$. Now, the variance may become minuscule compared to the size on n, so using algorithm (2) is very useful in terms of cost.

### 7. Summary

A quick overview to sum up the overall information discussed in this paper. The Secretary Problem is a very power tool that can be used in multiple different ways and be the fundamentals of a solution to other problems. The background and history of where it first appeared were discussed. It was shown that it can be solved in multiple ways as far as the algorithm used. K-Choice Secretary problem is another form of the Secretary Problem and how it was used to find an optimal r for multiple selections. Online Auction and the explanation of its type of environment and how the Secretary / K-Choice Secretary problem was a major tool in the online environment, giving an optimal solution. Models of each mentioned to show how they relate to one another and explanation of finding the probability and optimal r. Also, research in optimal comparison was studied between algorithms (1) and (2) by examining their results in multiple n tests.

### 8. References


