Lecture I - Optimal Stopping

Programming: Everyday Decision-Making Algorithms

Dr. Tobias Vlćek

Kühne Logistics University Hamburg - Winter 2025

Optimal Stopping

What is Optimal Stopping?

Question: Anybody know what optimal stopping is?

- Optimal stopping is the problem of:
  - ‣ choosing the best option
  - ‣ from a sequence of options
  - ‣ where the options are revealed one by one

Anybody an

example of

optimal stopping?

Flat Hunting

Photo by Aditya Ghosh on Unsplash

Hiring applicants

Photo by Scott Graham on Unsplash

Dating

Photo by Shelby Deeter on Unsplash

Searching for a parking spot

Photo by Joseph Pearson on Unsplash

"Secretary Problem"

The Secretary Problem

- Imagine you're hiring a secretary
- You must interview candidates one by one
- Now, you must decide: hire or continue searching
- Once you reject a candidate, you cannot go back
- How to maximize chance of selecting the best candidate?

. . .

> **i** Note
>
> The name is a bit misleading, as the problem is not about hiring a secretary, but about finding the best candidate. It comes from the 1960s and thus a little outdated.

Basic Setup

- We have n candidates
- We interview them one by one
- We must decide to hire or continue searching
- Ordinal ranking of candidates

. . .

Question: Anybody know what ordinal ranking is?

Ways to fail

Question: Anybody an idea how we can fail?

. . .

1. Reject all candidates and never hire - stopping too late
2. You hire someone too early - stopping too early

Ideas?

Look-and-Leap Strategy

The optimal strategy is to:

1. Look at the first 37 % of options
2. Remember the best one seen so far
3. Choose the next option that's better than the best seen
4. Chance of selecting the best candidate is 37 %[1]
5. Thus, we can fail with 63 %!

Step-by-step Approximation

- With each additional candidate, the chance of getting a new best candidate decreases
  - ‣ 1 candidate: 100%
  - ‣ 2 candidates: 50%
  - ‣ 3 candidates: 33%
  - ‣ 4 candidates: 25%
  - ‣ 5 candidates: 20%

. . .

Question: Anybody see a pattern?

---

[1]Large number of candidates! With a small number of candidates, we can do even better.

Why 37%?

- This is based on the geometric distribution
- The optimal stopping point is at n/e[2]
- e is the base of the natural logarithm (≈ 2.718)
- This comes from maximizing the probability of success

...

Computing the number

```python
import math

percentage = 1/math.e
print(f"Percentage of options to look at: {percentage:.3f}%")

candidates = 20
lookout_phase = candidates/math.e
print(f"Look at first {lookout_phase:.3f} candidates")
```

```
Percentage of options to look at: 0.368%
Look at first 7.358 candidates
```

...

> **ⓘ Note**
>
> No worries if you don't understand the code! We are essentialy just using the formula to calculate the percentage of candidates to look at.

Geometric Distribution

Let's visualize the success of a simulation with 20 candidates:

---

[2]This is a bit more advanced. We will not go into the details of the math here and focus more on the insights. For more details see Ferguson, T.S. (1989) 'Who solved the secretary problem?', Statistical Science, 4(3). doi:10.1214/ss/1177012493.

Secretary Problem Success Rate
(Based on 10,000 trials per position)
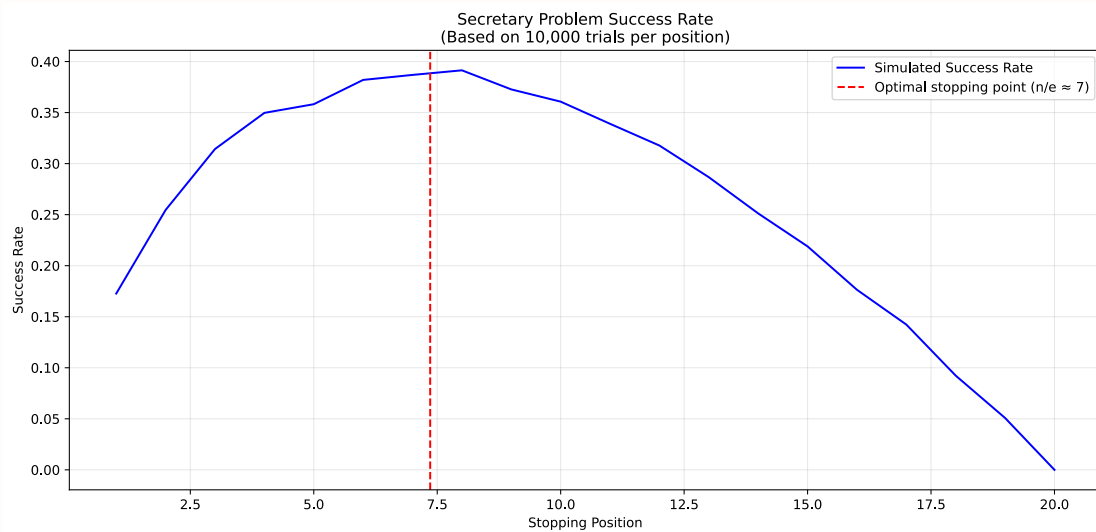
Variations

Rejection

Question: Imagine a dating scenario, where the other person can also reject you. Optimal stopping point?

- The optimal stopping point is now lower
- Because we can now fail more often
- With 50 % chance of rejection, we start leaping at 25 %
- Formula: $q^{\frac{1}{1-q}}$ with q being the chance of rejection

The role of time

What if we don't have a fixed number of candidates, but rather a fixed amount of time?

. . .

- Imagine we have one year to find a new flat
- We want the best flat, but don't know what a good flat is

. . .

Question: How should we behave?

. . .

- Same, but now we decide when to stop searching!

Other versions

- Selling a house for the best price ("Threshold Rule")
- Stealing with a success probability ("Burglar's Problem")
- Finding a parking spot ("Parking Lot Problem") [^2]

. . .

> **i Note**
>
> Side note for drivers: An increase in occupancy from 90 to 95% doubles the search time for all drivers!

Optimal Solution

The Gambler's Fallacy

Question: Can you imagine a scenario where it would be unwise to use optimal stopping?

. . .

- Imagine a game with a 50 % chance of winning
- If you win, your payoff is a triple of your bet
- If you lose, you have to pay your bet

Questions?

The End

> **i Note**
>
> That's it for todays lecture!
> We now have covered a brief introduction into optimal stopping and seen how to set up Python. Now we can start with the tutorials!

Literature

Interesting literature to start

- Christian, B., & Griffiths, T. (2016). Algorithms to live by: the computer science of human decisions. First international edition. New York, Henry Holt and Company.[3]
- Ferguson, T.S. (1989) 'Who solved the secretary problem?', Statistical Science, 4(3). doi:10.1214/ss/1177012493.

Books on Programming

- Downey, A. B. (2024). Think Python: How to think like a computer scientist (Third edition). O'Reilly. [Here](#)
- Elter, S. (2021). Schrödinger programmiert Python: Das etwas andere Fachbuch (1. Auflage). Rheinwerk Verlag.

. . .

---

[3]The main inspiration for this lecture. Nils and I have read it and discussed it in depth, always wanting to translate it into a course.

> **i Note**
>
> Think Python is a great book to start with. It's available online for free. Schrödinger Programmiert Python is a great alternative for German students, as it is a very playful introduction to programming with lots of examples.

More Literature

For more interesting literature, take a look at the [literature list](#) of this course.