Dating Problem via Linear Programming

Overview

Problem Description

Problem Description

Classical Dating Problem (Secretary Problem)

Problem Description

Classical Dating Problem (Secretary Problem)

- **Suppose** you are going to dating with several boys/girls one by one

Problem Description

Classical Dating Problem (Secretary Problem)

- **Suppose** you are going to dating with several boys/girls one by one
- These persons come in a uniform random order

Problem Description

Classical Dating Problem (Secretary Problem)

- **Suppose** you are going to dating with several boys/girls one by one
- These persons come in a uniform random order
- As you are a decent person, you will make decision immediately after dating with each boys/girls:

Problem Description

Classical Dating Problem (Secretary Problem)

- **Suppose** you are going to dating with several boys/girls one by one
- These persons come in a uniform random order
- As you are a decent person, you will make decision immediately after dating with each boys/girls:
    - If current person is chosen, you will not dating with unseen ones;

Problem Description

Classical Dating Problem (Secretary Problem)

- **Suppose** you are going to dating with several boys/girls one by one
- These persons come in a uniform random order
- As you are a decent person, you will make decision immediately after dating with each boys/girls:
  - If current person is chosen, you will not dating with unseen ones;
  - If current one is rejected, you can continue dating with next one, however, the rejected one will never come back.

Problem Description

Classical Dating Problem (Secretary Problem)

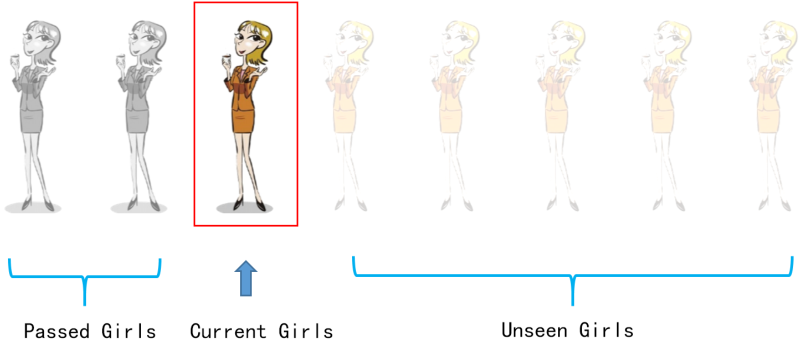
- **Suppose** you are going to dating with several boys/girls one by one
- These persons come in a uniform random order
- As you are a decent person, you will make decision immediately after dating with each boys/girls:
  - If current person is chosen, you will not dating with unseen ones;
  - If current one is rejected, you can continue dating with next one, however, the rejected one will never come back.
- Certainly, you want to select the best boy/girl as your soul mate

Problem Description

Classical Dating Problem (Secretary Problem)

- **Suppose** you are going to dating with several boys/girls one by one
- These persons come in a uniform random order
- As you are a decent person, you will make decision immediately after dating with each boys/girls:
  - If current person is chosen, you will not dating with unseen ones;
  - If current one is rejected, you can continue dating with next one, however, the rejected one will never come back.
- Certainly, you want to select the best boy/girl as your soul mate
- So, how to design a strategy to maximum the probability that the best one is chosen?

Example



Problem Analysis

Hardness of Dating Problem

Problem Analysis

Hardness of Dating Problem

- Lack of information: when you make decision for current person, you have no information about future ones

Problem Analysis

Hardness of Dating Problem

- Lack of information: when you make decision for current person, you have no information about future ones
- One-shot decision: Once you make a decision on a person, you have no chance to regret

Linear Programming Model

Linear Programming Model

Analysis

- Suppose there are n persons to date with

Linear Programming Model

Analysis

- Suppose there are n persons to date with
- In any *reasonable* algorithm, if you want to select a person as your soul mate, she/he must be the best one up to now

Linear Programming Model

Analysis

- Suppose there are n persons to date with
- In any *reasonable* algorithm, if you want to select a person as your soul mate, she/he must be the best one up to now
- Let x_i stand for the probability that the i-th one is chosen given she/he is the best person from position 1 to i

Linear Programming Model

Analysis

- Suppose there are n persons to date with
- In any *reasonable* algorithm, if you want to select a person as your soul mate, she/he must be the best one up to now
- Let x_i stand for the probability that the i-th one is chosen given she/he is the best person from position 1 to i
- That's to say

$$x_i = \Pr[\text{The } i\text{-th one is selected} \mid \text{The } i\text{-th one is the best one from 1 to } i] \quad (1)$$

Linear Programming Model

Analysis

- Suppose there are n persons to date with
- In any *reasonable* algorithm, if you want to select a person as your soul mate, she/he must be the best one up to now
- Let x_i stand for the probability that the i-th one is chosen given she/he is the best person from position 1 to i
- That's to say

$$x_i = \Pr[\text{The } i\text{-th one is selected} \mid \text{The } i\text{-th one is the best one from 1 to } i] \tag{1}$$

Remark

It should be emphasis that the probability x_i is defined on the randomness of coming sequence

Linear Programming Model

Analysis

- Clearly, we have

$$
\begin{aligned}
x_i &\leq \Pr[\text{No one is selected before position } i] \\
&= 1 - \sum_{l=1}^{i-1} \Pr[\text{The } l\text{-th one is selected}] \\
&= 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l
\end{aligned}
$$

Linear Programming Model

Analysis

- Let Z stand for the event that the best person is chosen

Linear Programming Model

Analysis

- Let Z stand for the event that the best person is chosen
- It is clearly that

$$
\begin{aligned}
\Pr[Z] &= \sum_{i=1}^{n} \Pr[\text{The } i\text{-th one is the best person and she/he is chosen}] \\
&= \sum_{i=1}^{n} \Pr[\text{The } i\text{-th one is the best person}] \\
&\quad \cdot \Pr[\text{The } i\text{-th one is selected} | \text{The } i\text{-th one is the best person}] \\
&= \sum_{i=1}^{n} \frac{1}{n} \cdot x_i
\end{aligned} \tag{2}
$$

Linear Programming Model

Primal

- In summary we have the following LP model:

$$\begin{aligned} \textbf{Maximize: } & \Pr[Z] = \sum_{i=1}^{n} x_i \\ \textbf{s.t. } & x_i \leq 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l, \quad 1 \leq i \leq n \\ & x_i \geq 0, \quad 1 \leq i \leq n. \end{aligned} \tag{3}$$

Linear Programming Model

Primal

- In summary we have the following LP model:

$$\begin{aligned}
&\textbf{Maximize:} \quad \Pr[Z] = \sum_{i=1}^{n} x_i \\
&\textbf{s.t.} \quad x_i \le 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l, \quad 1 \le i \le n \\
&\qquad\quad x_i \ge 0, \qquad\qquad\qquad\quad 1 \le i \le n.
\end{aligned} \tag{3}$$

- So simple a LP model!

Linear Programming Model
Dual

- Recall the relationship between the primal LP and its dual

Linear Programming Model

Dual

- Recall the relationship between the primal LP and its dual
- Let y_i stand for the dual variable of x_i

Linear Programming Model

Dual

- Recall the relationship between the primal LP and its dual
- Let y_i stand for the dual variable of x_i
- We have the following dual LP of the Dating Problem:

$$\begin{aligned} &\textbf{Minimize:} \quad \sum_{i=1}^{n} y_i \\ &\textbf{s.t.} \quad y_i + \frac{1}{i} \sum_{l=i+1}^{n} y_l \geq \frac{1}{n}, \quad 1 \leq i \leq n \\ &\qquad\quad y_i \geq 0, \qquad\qquad\qquad\quad 1 \leq i \leq n. \end{aligned} \tag{4}$$

Linear Programming Model

Dual

- Recall the relationship between the primal LP and its dual
- Let y_i stand for the dual variable of x_i
- We have the following dual LP of the Dating Problem:

$$
\begin{aligned}
\textbf{Minimize:} \quad & \sum_{i=1}^{n} y_i \\
\textbf{s.t.} \quad & y_i + \frac{1}{i} \sum_{l=i+1}^{n} y_l \geq \frac{1}{n}, \quad && 1 \leq i \leq n \\
& y_i \geq 0, && 1 \leq i \leq n.
\end{aligned}
\tag{4}
$$

- What's the meaning of y_i?

Linear Programming Model

Dual

- Recall the relationship between the primal LP and its dual
- Let y_i stand for the dual variable of x_i
- We have the following dual LP of the Dating Problem:

$$
\begin{aligned}
&\textbf{Minimize:} \quad \sum_{i=1}^{n} y_i \\
&\textbf{s.t.} \quad y_i + \frac{1}{i}\sum_{l=i+1}^{n} y_l \geq \frac{1}{n}, \quad 1 \leq i \leq n \\
&\qquad\quad y_i \geq 0, \qquad\qquad\qquad\quad 1 \leq i \leq n.
\end{aligned} \tag{4}
$$

- What's the meaning of y_i?
- It is an open problem. Clever as you may have some idea.

Linear Programming Model

Strictness of the LP Model

- Although simple, the primal LP *strictness* for Dating Problem

Linear Programming Model

Strictness of the LP Model

- Although simple, the primal LP *strictness* for Dating Problem
- *Strictness* means any algorithm can be mapped to a feasible solution of the LP and any solution of the LP can be mapped to an algorithm

Linear Programming Model

Strictness of the LP Model

- Although simple, the primal LP *strictness* for Dating Problem
- *Strictness* means any algorithm can be mapped to a feasible solution of the LP and any solution of the LP can be mapped to an algorithm
- Why strictness is important?

Linear Programming Model

Strictness of the LP Model

- Although simple, the primal LP *strictness* for Dating Problem
- *Strictness* means any algorithm can be mapped to a feasible solution of the LP and any solution of the LP can be mapped to an algorithm
- Why strictness is important?
- Only when a LP is strict, the optimal solution of the LP can be an optimal solution for the problem

Linear Programming Model

Strictness of the LP Model

- It easy to check that any algorithm π can be mapped to a feasible solution of the primal LP:

Linear Programming Model

Strictness of the LP Model

- It easy to check that any algorithm π can be mapped to a feasible solution of the primal LP:
    - We can calculate the value x_i in π

Linear Programming Model

Strictness of the LP Model

- It easy to check that any algorithm π can be mapped to a feasible solution of the primal LP:
  - We can calculate the value x_i in π
  - Then, we can check that these x_is obtained satisfies the constraint

Linear Programming Model

Strictness of the LP Model

- It easy to check that any algorithm π can be mapped to a feasible solution of the primal LP:
  - We can calculate the value x_i in π
  - Then, we can check that these x_is obtained satisfies the constraint
  - We omit this proof here. (Maybe a homework?)

Linear Programming Model

Strictness of the LP Model

- It easy to check that any algorithm π can be mapped to a feasible solution of the primal LP:
  - We can calculate the value x_i in π
  - Then, we can check that these x_is obtained satisfies the constraint
  - We omit this proof here. (Maybe a homework?)
- We should show that any feasible solution can be mapped to an algorithm for Dating Problem

Linear Programming Model

Strictness of the LP Model

- Suppose $\{x_i' | 1 \leq i \leq n\}$ is a set of feasible solution of the primal LP

Linear Programming Model

Strictness of the LP Model

- Suppose $\{x_i' | 1 \leq i \leq n\}$ is a set of feasible solution of the primal LP
- We construct an algorithm π based on x_i' as follow:

Linear Programming Model

Strictness of the LP Model

- Suppose $\{x'_i | 1 \leq i \leq n\}$ is a set of feasible solution of the primal LP
- We construct an algorithm π based on x'_i as follow:
  - For each person j, if she/he is the best one up to now, select is with probability $\frac{x_i}{1-\sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l}$

Linear Programming Model

Strictness of the LP Model

- Suppose $\{x_i'|1 \leq i \leq n\}$ is a set of feasible solution of the primal LP
- We construct an algorithm π based on x_i' as follow:
  - For each person j, if she/he is the best one up to now, select is with probability $\frac{x_i}{1-\sum_{l=1}^{i-1}\frac{1}{l}\cdot x_l}$
  - Otherwise, that's j is not the best one up to now, discard she/him

Linear Programming Model

Strictness of the LP Model

- Suppose $\{x_i' | 1 \leq i \leq n\}$ is a set of feasible solution of the primal LP
- We construct an algorithm π based on x_i' as follow:
  - For each person j, if she/he is the best one up to now, select is with probability $\frac{x_i}{1-\sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l}$
  - Otherwise, that's j is not the best one up to now, discard she/him
- It's over?

- Suppose $\{x_i'|1 \leq i \leq n\}$ is a set of feasible solution of the primal LP
- We construct an algorithm π based on x_i' as follow:
  - For each person j, if she/he is the best one up to now, select is with probability $\frac{x_i}{1-\sum_{l=1}^{i-1}\frac{1}{l}\cdot x_l}$
  - Otherwise, that's j is not the best one up to now, discard she/him
- It's over?
- Naive! We should show that $x_i(\pi) = x_i'$, where $x_i(\pi)$ is the probability that i-th one is selected give she/he is the best one up to i in algorithm π

Linear Programming Model

Strictness of the LP Model

- Suppose $\{x'_i | 1 \leq i \leq n\}$ is a set of feasible solution of the primal LP
- We construct an algorithm π based on x'_i as follow:
  - For each person j, if she/he is the best one up to now, select is with probability $\frac{x_i}{1-\sum_{l=1}^{i-1}\frac{1}{l}\cdot x_l}$
  - Otherwise, that's j is not the best one up to now, discard she/him
- It's over?
- Naive! We should show that $x_i(\pi) = x'_i$, where $x_i(\pi)$ is the probability that i-th one is selected give she/he is the best one up to i in algorithm π
- Think about why we should show this and show it by yourself (May a homework?)

Optimal Algorithm

Optimal Algorithm
Randomized

- Somehow we have solved the Dating Problem:

Optimal Algorithm
Randomized

- Somehow we have solved the Dating Problem:
  - Solve the primal LP

Optimal Algorithm
Randomized

- Somehow we have solved the Dating Problem:
  - Solve the primal LP
  - Construct an algorithm based on the optimal solution of LP as previous slides

Optimal Algorithm
Randomized

- Somehow we have solved the Dating Problem:
  - Solve the primal LP
  - Construct an algorithm based on the optimal solution of LP as previous slides
  - Finally, we have an optimal randomized algorithm

Optimal Algorithm

Randomized

- Somehow we have solved the Dating Problem:
  - Solve the primal LP
  - Construct an algorithm based on the optimal solution of LP as previous slides
  - Finally, we have an optimal randomized algorithm
- It is not over so far

Optimal Algorithm

Deterministic

- In fact, there is an optimal deterministic algorithm for the Dating Problem:

Optimal Algorithm

Deterministic

- In fact, there is an optimal deterministic algorithm for the Dating Problem:
  - For the first $\frac{n}{e}$ persons, you just dating but select nobody

Optimal Algorithm

Deterministic

- In fact, there is an optimal deterministic algorithm for the Dating Problem:
  - For the first $\frac{n}{e}$ persons, you just dating but select nobody
  - For the rest persons, select the first one that better than all previous

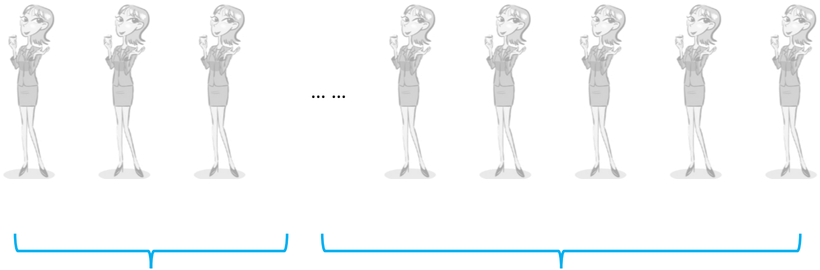
Optimal Algorithm

Deterministic

- In fact, there is an optimal deterministic algorithm for the Dating Problem:
  - For the first $\frac{n}{e}$ persons, you just dating but select nobody
  - For the rest persons, select the first one that better than all previous
- By this algorithm, you can find your soul mate with probability $\frac{1}{e}$

Optimal Algorithm

Deterministic

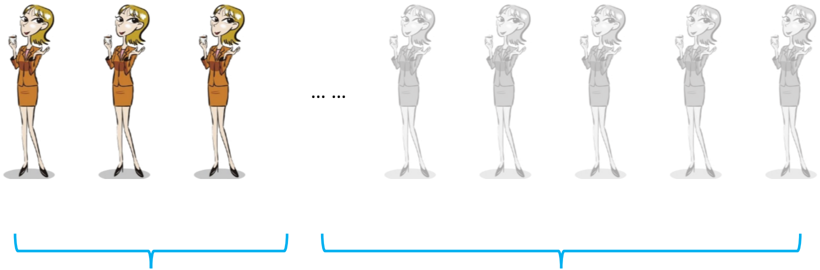


For the first $\frac{n}{e}$ persons, no one is selected

In this phase, if any one is the best person up to now, select she/him

Optimal Algorithm

Deterministic

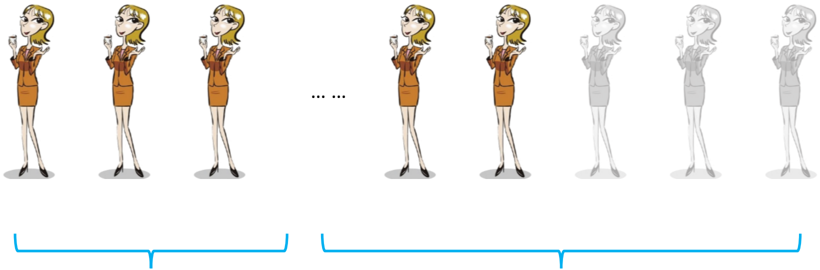


For the first $\frac{n}{e}$ persons, no one is selected

In this phase, if any one is the best person up to now, select she/him

Optimal Algorithm

Deterministic

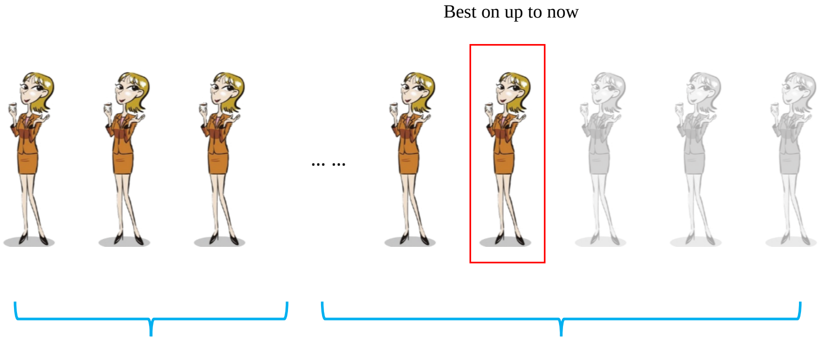


For the first $\frac{n}{e}$ persons, no one is selected

In this phase, if any one is the best person up to now, select she/him

Optimal Algorithm

Deterministic



For the first $\frac{n}{e}$ persons, no one is selected

In this phase, if any one is the best person up to now, select she/him

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies
  - $x_i = 0$, for $1 \leq i \leq \frac{n}{e}$

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies
  - $x_i = 0$, for $1 \leq i \leq \frac{n}{e}$
  - $x_i = 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l$, otherwise

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies
  - $x_i = 0$, for $1 \leq i \leq \frac{n}{e}$
  - $x_i = 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l$, otherwise
- This is easy to check

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies
  - $x_i = 0$, for $1 \leq i \leq \frac{n}{e}$
  - $x_i = 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l$, otherwise
- This is easy to check
- By solve some small case using GLPK, we can find the optimal solution is exactly in this pattern

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies
  - $x_i = 0$, for $1 \leq i \leq \frac{n}{e}$
  - $x_i = 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l$, otherwise
- This is easy to check
- By solve some small case using GLPK, we can find the optimal solution is exactly in this pattern
- How to show that for general n the deterministic algorithm is optimal?

Optimal Algorithm

Deterministic

- This deterministic algorithm can be mapped to an feasible solution $\{x_i | 1 \leq i \leq n\}$ of the primal LP that satisfies
  - $x_i = 0$, for $1 \leq i \leq \frac{n}{e}$
  - $x_i = 1 - \sum_{l=1}^{i-1} \frac{1}{l} \cdot x_l$, otherwise
- This is easy to check
- By solve some small case using GLPK, we can find the optimal solution is exactly in this pattern
- How to show that for general n the deterministic algorithm is optimal?
- We will show the optimality by the Theorem of Complementary Slackness

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP
- Fixed the constraint of the dual LP tight, that's let

$$y_i + \frac{1}{i}\sum_{l=i+1}^{n} y_l = \frac{1}{n}$$

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP
- Fixed the constraint of the dual LP tight, that's let

$$y_i + \frac{1}{i} \sum_{l=i+1}^{n} y_l = \frac{1}{n}$$

- Solve this recursion from n to 1

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP
- Fixed the constraint of the dual LP tight, that's let

$$y_i + \frac{1}{i} \sum_{l=i+1}^{n} y_l = \frac{1}{n}$$

- Solve this recursion from n to 1
- We can get that $y_i = \frac{1}{n}\left(1 - \sum_{l=i}^{n-1} \frac{1}{l}\right)$

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP
- Fixed the constraint of the dual LP tight, that's let

$$y_i + \frac{1}{i} \sum_{l=i+1}^{n} y_l = \frac{1}{n}$$

- Solve this recursion from n to 1
- We can get that $y_i = \frac{1}{n} \left(1 - \sum_{l=i}^{n-1} \frac{1}{l} \right)$
- Clearly, y_i is decrease as i going down

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP
- Fixed the constraint of the dual LP tight, that's let

$$y_i + \frac{1}{i}\sum_{l=i+1}^{n} y_l = \frac{1}{n}$$

- Solve this recursion from n to 1
- We can get that $y_i = \frac{1}{n}\left(1 - \sum_{l=i}^{n-1}\frac{1}{l}\right)$
- Clearly, y_i is decrease as i going down
- When n is large enough we have $y_i \approx \frac{1-logn/i}{n}$ and it equals to 0 exactly when $i = \frac{n}{e}$!

Optimal Algorithm

Feasible Solution of Dual LP

- We first construct an feasible solution of the dual LP
- Fixed the constraint of the dual LP tight, that's let

$$y_i + \frac{1}{i}\sum_{l=i+1}^{n} y_l = \frac{1}{n}$$

- Solve this recursion from n to 1
- We can get that $y_i = \frac{1}{n}\left(1 - \sum_{l=i}^{n-1}\frac{1}{l}\right)$
- Clearly, y_i is decrease as i going down
- When n is large enough we have $y_i \approx \frac{1 - logn/i}{n}$ and it equals to 0 exactly when $i = \frac{n}{e}$!
- According to the non-negative constraint we set y_i to be 0 when $1 \leq i \leq \frac{n}{e}$

Optimal Algorithm

Feasible Solution of the Primal and Dual LP

- In summary, we have a feasible solution of the dual LP that satisfies:

$$y_i = \begin{cases} 0, & 1 \leq i \leq n/e \\ \dfrac{1}{n} - \dfrac{1}{i} \displaystyle\sum_{l=i+1}^{n} y_l & n/e < i \leq n \end{cases} \tag{5}$$

Optimal Algorithm

Feasible Solution of the Primal and Dual LP

- In summary, we have a feasible solution of the dual LP that satisfies:

$$y_i = \begin{cases} 0, & 1 \le i \le n/e \\ \frac{1}{n} - \frac{1}{i}\sum_{l=i+1}^{n} y_l & n/e < i \le n \end{cases} \tag{5}$$

- The deterministic algorithm can be mapped to an feasible solution of the primal LP that satisfies:

$$x_i = \begin{cases} 0, & 1 \le i \le n/e \\ 1 - \sum_{l=i+1}^{n} \frac{1}{l} x_l & n/e < i \le n \end{cases} \tag{6}$$

Optimal Algorithm

Slackness Variable of the Primal and Dual LP

- Let xs_i and ys_i stand for the corresponding slackness variable of x_i and y_i, we have

$$xs_i \begin{cases} \geq 0, & 1 \leq i \leq n/e \\ = 0 & n/e < i \leq n \end{cases} \tag{7}$$

$$ys_i \begin{cases} \geq 0, & 1 \leq i \leq n/e \\ = 0 & n/e < i \leq n \end{cases} \tag{8}$$

Optimal Algorithm

Slackness Variable of the Primal and Dual LP

- Let xs_i and ys_i stand for the corresponding slackness variable of x_i and y_i, we have

$$xs_i \begin{cases} \geq 0, & 1 \leq i \leq n/e \\ = 0 & n/e < i \leq n \end{cases} \tag{7}$$

$$ys_i \begin{cases} \geq 0, & 1 \leq i \leq n/e \\ = 0 & n/e < i \leq n \end{cases} \tag{8}$$

- Easy to check that those tow feasible solutions satisfy the Theorem of Complementary Slackness

Thanks!