Masterclass: Random uniform permutations (Nancy, June 2022)

Lucas Gerin, École Polytechnique (Palaiseau, France)
lucas.gerin@polytechnique.edu

This course is at the interplay between Probability and Combinatorics. It is intended for Master students with a background in Probability (random variables, expectation, conditional probability).

The question we will adress is "What can we say about a *typical* large permutation?": the number of cycles, their lengths, the number of fixed points,... This is also a pretext to present some universal phenomena in Probability: reinforcement, the Poisson paradigm, size-bias,...

Contents

Brief reminder on permutations

Before we turn to *random* permutations, we will give a few definitions regarding non-random (or *deterministic* permutations).

A *permutation* of size $n \geq 1$ is a bijection $\sigma : \{1, 2, \dots, n\} \to \{1, 2, \dots, n\}$. For example

$$\begin{matrix} 1 & 2 & 3 & 4 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 4 & 3 & 1 \end{matrix}$$

is a permutation of size 4. In these notes we often write a permutation with its one-line representation $\sigma(1)\sigma(2)\dots\sigma(n)$. For example the above permutation is simply written 2431.

There are $n!$ permutations of size n.

Cycle decomposition

For our purpose, there is a convenient alternative way to encode a permutation: by its *cycle decomposition*. A *cycle* is a finite sequence of distinct integers, defined up to the cycle order. This means that the three following denote the same cycle:

$$(8,3,4) = (3,4,8) = (4,8,3),$$

while $(8,3,4) \neq (8,4,3)$.

The *cycle decomposition* of a permutation σ is defined as follows. We give the theoretical algorithm and detail the example of this permutation of size 7:

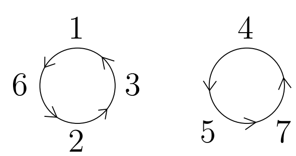
$$\begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 6 & 3 & 1 & 5 & 7 & 2 & 4 \end{array}$$

Algorithm	**Example**
Start with 1st cycle (1)	(1)
Add to this cycle $\sigma(1)$, then $\sigma(\sigma(1))$, then $\sigma(\sigma(\sigma(1)))$, and so one until one of these numbers is equal to 1.	$(1) \to (1,6) \to (1,6,2) \to (1,6,2,3)$ and the cycle is over since $\sigma(3) = 1$.
Start the 2d cycle with a number which has not been seen before.	1st cycle $(1,6,2,3)$. 2d cycle: (4)
Complete the 2d cycle with same procedure.	1st cycle $(1,6,2,3)$. 2d cycle: $(4) \to (4,5) \to (4,5,7)$.
Create new cycles until there is no remaining number.	Done.

Finally, the cycle decomposition of σ is

$$(1,6,2,3),(4,5,7)$$

It is convenient to represent the cycle decomposition of σ with the following diagram:



Exercise 1 What is the cycle decomposition of 62784315?

Remark . *By construction the cycle decomposition is unique, up to a rearrangement of cycles. For instance*

$$(1,6,2,3),(4,5,7) \text{ and } (4,5,7),(1,6,2,3)$$

describe the same permutation. A way to ensure uniqueness is to order cycles by the increasing order of their smallest elements.

1 How to simulate a random uniform permutation?

We will first discuss the following question. Imagine that you are given a random number generator `rand` (in your favourite programming language) which returns independent uniform random variables. How to use `rand` to simulate a random uniform permutation of size n?

1.1 The naive algorithm

It works as follows:

- Pick $\sigma(1)$ uniformly at random in $\{1, 2, \ldots, n\}$ (n choices);
- Pick $\sigma(2)$ uniformly at random in $\{1, 2, \ldots, n\} \setminus \{\sigma(1)\}$ ($n-1$ choices);
- Pick $\sigma(3)$ uniformly at random in $\{1, 2, \ldots, n\} \setminus \{\sigma(1), \sigma(2)\}$ ($n-2$ choices),

and so on until $\sigma(n)$ (1 choice).

By construction every permutation occurs with probability $1/n!$ so the output is uniform.

1.2 The "continuous" algorithm

- Pick continuous i.i.d. random variables $X_1, X_2, \ldots, X_n$ with some density f ;
- With probability one the n values are pairwise distinct (see the proof below). Therefore there exists a unique permutation σ such that

$$X_{\sigma(1)} < X_{\sigma(2)} < X_{\sigma(3)} < \cdots < X_{\sigma(n)}.$$

- This σ is your output.

Proposition 1. *For every n, the output of the continuous algorithm is uniform among the $n!$ permutations of size n.*

Proof.
(We do the proof in the case where X_i's are uniform in $(0,1)$.)
Step 1: The n values are distinct. We have to prove that

$$\mathbb{P}\left(\text{ for all } i \neq j,\ X_i \neq X_j \right) = 1.$$

We prove that the complement event {there are i, j such that $X_i = X_j$} has probability zero. First we notice that

$$\mathbb{P}(\text{ there are } i \neq j \text{ such that } X_i = X_j) = \mathbb{P}\left(\cup_{i \neq j} \{X_i = X_j\}\right) \leq \sum_{i \neq j} \mathbb{P}\left(X_i = X_j\right),$$

by the union bound[(i)]. Now,

$$\mathbb{P}\left(X_i = X_j\right) = \int_{(0,1)^2} \mathbf{1}_{x=y} dx dy = \int_{y \in (0,1)} \left(\int_{x \in (0,1)} \mathbf{1}_{x=y} dx \right) dy = \int_{y \in (0,1)} \left(\int_{x=y}^{y} dx \right) dy = \int_{y \in (0,1} 0 \times dy = 0.$$

Step 2: The output σ is uniform. To avoid messy notations we make the proof in the case $n = 3$. Since the 3 values X_1, X_2, X_3 are distinct we have

$$\begin{aligned}
1 &= \mathbb{P}(X_1 < X_2 < X_3) + \mathbb{P}(X_1 < X_3 < X_2) + \mathbb{P}(X_2 < X_1 < X_3) \\
&\quad + \mathbb{P}(X_2 < X_3 < X_1) + \mathbb{P}(X_3 < X_1 < X_2) + \mathbb{P}(X_3 < X_2 < X_1) \\
&= \int_{(0,1)^3} \mathbf{1}_{x_1 < x_2 < x_3} dx_1 dx_2 dx_3 + \int_{(0,1)^3} \mathbf{1}_{x_1 < x_3 < x_2} dx_1 dx_2 dx_3 + \int_{(0,1)^3} \mathbf{1}_{x_2 < x_1 < x_3} dx_1 dx_2 dx_3 \\
&\quad + \int_{(0,1)^3} \mathbf{1}_{x_2 < x_3 < x_1} dx_1 dx_2 dx_3 + \int_{(0,1)^3} \mathbf{1}_{x_3 < x_1 < x_2} dx_1 dx_2 dx_3 + \int_{(0,1)^3} \mathbf{1}_{x_3 < x_2 < x_1} dx_1 dx_2 dx_3.
\end{aligned}$$

Now, x_1, x_2, x_3 are dummy variables in the above integrals, so they are interchangeable. Therefore, these 6 integrals are identical and each of these is $1/6 = 1/3!$. □

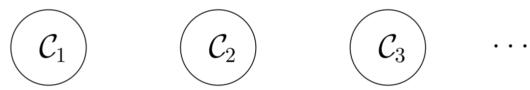
[(i)] The union bound says that $\mathbb{P}\left(\bigcup_{n \geq 1} A_n\right) \leq \sum_{n \geq 1} \mathbb{P}(A_n)$ for every sequence of events (A_n).

1.3 The "Chinese restaurant" algorithm

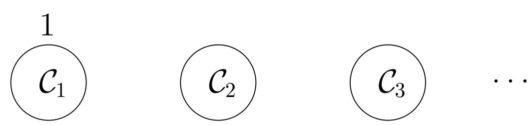
We introduce the Chinese restaurant algorithm, also called the Fisher-Yates algorithm (or even Fisher-Yates-Knuth algorithm). The main difference with the two previous algorithms is that the output σ will be described through its cycle decomposition.

The algorithm runs as follows:

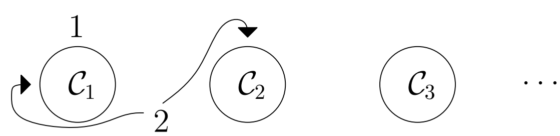
- Assume we are given infinitely many "restaurant tables" $\mathcal{C}_1, \mathcal{C}_2, \ldots$. These tables are large enough so that an arbitrary number of people can sit at each table.



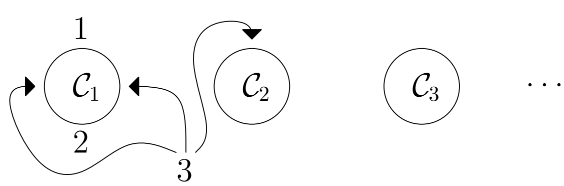
- Infinitely many customers $1, 2, 3, \ldots$ enter the restaurant, one at a time. Put Customer n.1 at table $\mathcal{C}_1$:



- With equal probability one-half, put Customer n.2 either at the same table as 1 (on its right) or alone at the new table $\mathcal{C}_2$:

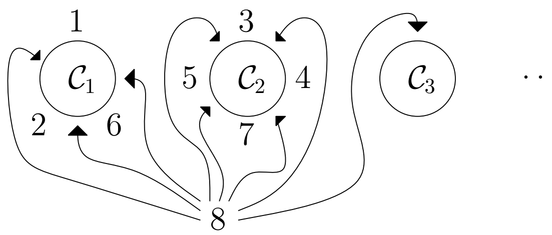


- With equal probability one-third, put Customer n.3 either on the right of 1, or on the right of 2, or alone at the first empty table:

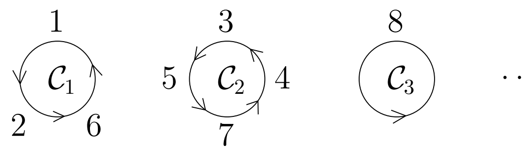


- ...

- Assume that customers $1, 2, \ldots, n-1$ are already installed. With equal probability $1/n$, put Customer n either on the right of 1, ..., or on the right of $n-1$, or alone at the first empty table (here $n = 8$):



Now, we return the permutation σ whose cycle decomposition corresponds to table repartitions. Assume here that 8 sits alone, we obtain the diagram



This can also be written (126)(3547)(8). The corresponding permutation is

$$\begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 6 & 5 & 3 & 7 & 1 & 4 & 8 \end{array}$$

Proposition 2. *For every n, the output of the Chinese restaurant algorithm is uniform among the $n!$ permutations of size n.*

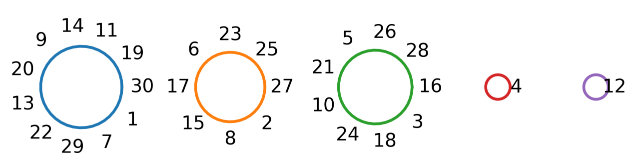
Proof. By construction, each table repartition with n customers occurs with the same probability

$$1 \times \frac{1}{2} \times \frac{1}{3} \times \cdots \times \frac{1}{n}.$$

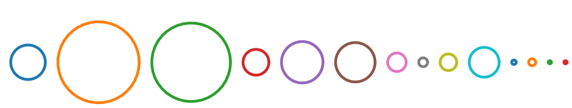
Now, each table repartition corresponds to exactly one permutation of size n. Therefore each permutation occurs with probability $1/n!$. □

Simulations

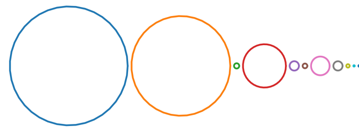
Here is a simulation for $n = 30$:



Here is a simulation for $n = 2000$ (We only represent sizes of tables. They have respective sizes $122, 673, 631, 68, 176, 159, 35, 8, 28, 91, 2, 5, 1, 1.$):



A last simulation for $n = 30000$. Tables have sizes $15974, 11238, 31, 2121, 99, 25, 397, 97, 13, 2, 3$.



For more on the Chinese restaurant we refer to [5]. On the following webpage you can run simulations of the Chinese restaurant by yourself:

`http://gerin.perso.math.cnrs.fr/Enseignements/ChineseRestaurant.html`

2 Typical properties of a random uniform permutation

Frow now on S_n denotes a random uniform permutation of size n, generated by any of the previous algorithms.

2.1 Number of fixed points

Definition 1. *Let σ be a permutation of size n. The integer $1 \le i \le n$ is a fixed point of σ if $\sigma(i) = i$.*

For example, 2431 has a unique fixed point at $i = 3$.

Proposition 3. *Let F_n be the number of fixed points of S_n. For every n, we have that*[(ii)]

$$\mathbb{E}[F_n] = 1, \qquad \mathrm{Var}(F_n) = 1.$$

This is quite surprising that $\mathbb{E}[F_n]$ and $\mathrm{Var}(F_n)$ do not depend on n.

Proof. We write $F_n = \sum_{i=1}^n X_i$, where

$$X_i = \begin{cases} 1 & \text{if } S_n(i) = i, \\ 0 & \text{otherwise} \end{cases}.$$

Random variables X_i's are *not* independent. Still we have by linearity of expectation that

$$\mathbb{E}[F_n] = \mathbb{E}[X_1 + \cdots + X_n] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_n],$$

and we are left to compute $\mathbb{E}[X_i]$ for every i. Now,

$$\mathbb{P}(X_i = 1) = \mathbb{P}(S_n(i) = i) = \frac{\mathrm{card}\,\{\text{permutations } s \text{ of size } n \text{ with } s(i) = i\}}{\mathrm{card}\,\{\text{permutations of size } n\}} = \frac{(n-1)!}{n!} = \frac{1}{n}.$$

(Indeed, a permutation such that $s(i) = i$ is also a permutation of the set $\{1, 2, \ldots, i-1, i+1, \ldots, n\}$ of size $n - 1$.) Therefore we have that

$$\mathbb{E}[X_i] = 1 \times \mathbb{P}(X_i = 1) + 0 \times \mathbb{P}(X_i = 0) = 1/n.$$

(ii) Thank you to Amic Frouvelle for pointing me that the variance was wrong in the previous version of these notes.

Finally

$$\mathbb{E}[F_n] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_n] = n \times 1/n = 1.$$

In order to compute the variance we will use the formula

$$\begin{aligned}\mathrm{Var}(\sum X_i) &= \sum_i \mathrm{Var}(X_i) + \sum_{i \neq j} \mathrm{Cov}(X_i, X_j) \\ &= n\mathrm{Var}(X_1) + n(n-1)\mathrm{Cov}(X_1, X_2)\end{aligned}$$

By the previous computation we have:

$$\mathbb{E}[X_1] = \tfrac{1}{n}, \qquad \mathrm{Var}(X_1) = \tfrac{1}{n}(1 - \tfrac{1}{n}).$$

Similarly as above we can compute

$$\mathbb{E}[X_1X_2] = \mathbb{P}(X_1 \times X_2 = 1) = \mathbb{P}(X_1 = 1, X_2 = 1) = \frac{(n-2)!}{n!} = \frac{1}{n(n-1)}.$$

Hence $\mathrm{Cov}(X_1, X_2) = \frac{1}{n(n-1)} - \mathbb{E}[X_1]\mathbb{E}[X_2] = \frac{1}{n(n-1)} - \frac{1}{n^2}$. Finally

$$\mathrm{Var}(F_n) = 1 - \frac{1}{n} + n(n-1)\frac{1}{n^2(n-1)} = 1.$$

□

The Poisson paradigm

There is a general phenomenon in probability known as the *Poisson paradigm*. It says that if X_i's are 0/1 random variable such that

1. $\mathbb{E}[X_i] = \mathbb{P}(X_i = 1)$ is "small" for every i ;
2. X_i's are "almost" independent ;

then $X = \sum X_i$ is almost distributed like the Poisson distribution with mean $\sum \mathbb{E}[X_i]$. Here $\sum \mathbb{E}[X_i] = \sum_{i=1}^n 1/n = 1$ and one can make the Poisson paradigm rigorous:

Proposition 4 (See [8])**.** *Let* $(S_n)_n$ *be a sequence of random uniform permutations, and let* F_n *be the number of fixed points of* S_n*. Then* F_n *converges* in distribution *to the Poisson distribution with mean* 1, i.e.

$$\mathbb{P}(F_n = k) \stackrel{n \to +\infty}{\to} \mathbb{P}(\text{Poisson}(1) = k) = \frac{e^{-1}}{k!},$$

for every $k = 0, 1, 2, \ldots$.

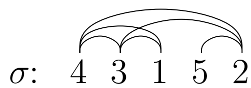
For more on the Poisson paradigm, we refer to [2].

2.2 Number of inversions

An *inversion* in σ is a pair (i,j) such that

$$\begin{cases} i < j, \\ \sigma(i) > \sigma(j) \end{cases}.$$

Let $\mathrm{Inv}(\sigma)$ be the number of inversions of σ. For example, if $\sigma = 43152$ then $\mathrm{Inv}(\sigma) = 6$ (each arc counts for an inversion):



Proposition 5. *For every n, let S_n be a uniform random permutation of size n. Then*

$$\mathbb{E}[\mathrm{Inv}_n(S_n)] = \frac{n(n-1)}{4}.$$

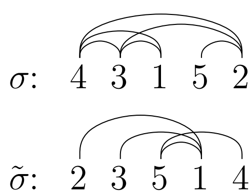
Proof. We will make a combinatorial proof, with (almost) no computation. First, let $\tilde{\sigma}$ be the *reversed* permutation of σ: for every $1 \leq i \leq n$,

$$\tilde{\sigma}(i) = n + 1 - \sigma(i).$$

For instance, if $\sigma = 43152$ then $\tilde{\sigma} = 23514$. Then by construction we have that an arbitrary pair (i,j) is an inversion for σ if and only if it is not an inversion for $\tilde{\sigma}$. We deduce that

$$\mathrm{Inv}(\sigma) + \mathrm{Inv}(\tilde{\sigma}) = \mathrm{card}\{\text{ all pairs } 1 \leq i < j \leq n\} = \binom{n}{2} = \frac{n(n-1)}{2}.$$

Here we see that $\mathrm{Inv}(43152) + \mathrm{Inv}(23514) = 6 + 4 = \binom{5}{2}$:



Now, we apply the above equality to $\sigma = S_n$ and take expectations of both sides:

$$\mathbb{E}\big[\mathrm{Inv}(S_n)\big] + \mathbb{E}\big[\mathrm{Inv}(\tilde{S_n})\big] = \frac{n(n-1)}{2}.$$

But now, it is obvious that $\sigma \mapsto \tilde{\sigma}$ is a bijection so it preserves the uniform measure. Therefore $\tilde{S_n}$ is also a uniform random permutation and we have $\mathbb{E}\big[\mathrm{Inv}(S_n)\big] = \mathbb{E}\big[\mathrm{Inv}(\tilde{S_n})\big]$. The proof is done. $\square$

2.3 Number of cycles

Proposition 6. *Let C_n be the number of cycles of S_n. When $n \to +\infty$,*

$$\mathbb{E}[C_n] \overset{n\to+\infty}{\sim} \log(n).$$

Proof. We may assume that S_n is the output of the Chinese restaurant algorithm. All along the process of the Chinese restaurant, a new cycle appears when a customer sits at a new table:

$$C_n = \sum_{i=1}^{n} Z_i,$$

where

$$Z_i = \begin{cases} 1 & \text{if Customer } i \text{ sits at a new table,} \\ 0 & \text{otherwise} \end{cases}.$$

Customer i sits at a new table with probability $1/i$, therefore $\mathbb{E}[Z_i] = 1/i$. Then,

$$\mathbb{E}[C_n] = \mathbb{E}\left[\sum_{i=1}^{n} Z_i\right] = \sum_{i=1}^{n} \mathbb{E}[Z_i] = \sum_{i=1}^{n} \frac{1}{i}.$$

Now, we use the fact that[(iii)] $\sum_{i=1}^{n} \frac{1}{i} \sim \log(n)$. □

Remark . *Random variables (Z_i) are actually independent (each customer sits at a new table, no matter what happened before). Thus we can easily calculate the variance:*

$$\mathrm{Var}(C_n) = \sum_{i=1}^{n} \mathrm{Var}(Z_i) = \sum_{i=1}^{n} \frac{1}{i}(1 - \frac{1}{i}) \sim \log(n).$$

2.4 Size of the first cycle/first table

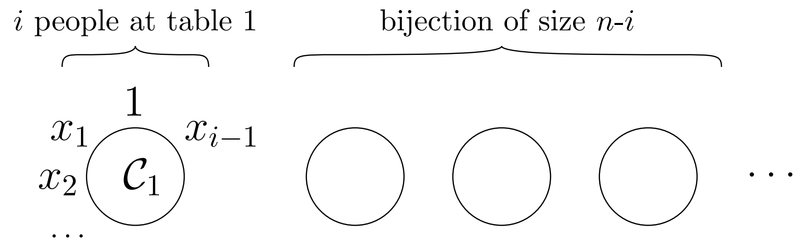
Let $T_1(n)$ be the number of customers at Table 1 in the Chinese restaurant process at time n. By Proposition 2, we have that the random variable $T_1(n)$ has the distribution of the cycle of 1 in the cycle decomposition of a random uniform permutation of size n.

Proposition 7. *For every n, the random variable $T_1(n)$ is uniformly distributed in $\{1, 2, \dots, n\}$,* i.e.

$$\mathbb{P}(T_1(n) = i) = \frac{1}{n}, \qquad \textit{for every } i \in \{1, 2, \dots, n\}.$$

Proof. For $i = 1, \dots, n$, let us enumerate the permutations in which $T_1(n) = i$. We have to choose $i-1$ elements $x_1, \dots, x_{i-1}$ ($\binom{n-1}{i-1}$ choices) which belong to this cycle, and put them in a given order ($(i-1)!$ choices). Then, the $n-i$ remaining elements form a permutation of size $n-i$ ($(n-i)!$ choices).

(iii) See https://en.wikipedia.org/wiki/Harmonic_series_(mathematics)



Therefore

$$\begin{aligned}\mathbb{P}(T_1(n) = i) &= \frac{\text{card}\left\{\text{permutations of size } n \text{ with } T_1(n) = i\right\}}{n!} \\ &= \frac{1}{n!}\binom{n-1}{i-1}(i-1)!(n-i)! \\ &= \frac{1}{n!}\frac{(n-1)!}{(i-1)!(n-i)!}(i-1)!(n-i)! = \frac{1}{n}.\end{aligned}$$

□

3 The Chinese restaurant process

We already saw that in order to study the properties of S_n it may be useful to consider that S_n is the output of the Chinese restaurant process. Let us show some more applications.

3.1 Size of the first cycle/first table (revisited)

We first provide another proof of Proposition 7 using the Chinese restaurant process. The idea is to look at the process $(T_1(n))_{n\geq 1}$, which is actually known as the *Pólya Urn process* [6].

Proof. The proof goes by induction. For $n = 1$ this is obvious since with probability one $T_1(1) = 1$.

Assume now that for some $n \geq 1$, the random variable $T_1(n)$ is uniform in $\{1, 2, \ldots, n\}$. If $T_1(n) = i$, then Customer $n + 1$ sits at table 1 with probability $i/(n + 1)$.

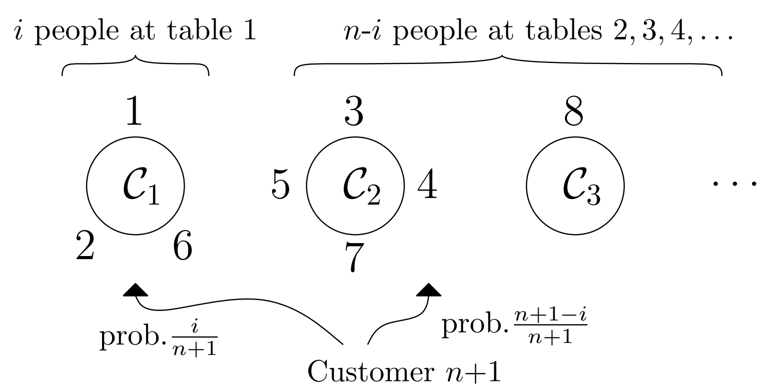


Figure: *A sketch of the situation when Customer $n + 1$ tries to sit.*

Therefore

$$T_1(n+1) = \begin{cases} i+1 & \text{with probab. } \frac{i}{n+1}, \\ i & \text{with probab. } \frac{n+1-i}{n+1}. \end{cases} \tag{1}$$

Fix $j \in \{1, \ldots, n+1\}$. The above argument implies that

$$\begin{aligned}
\mathbb{P}(T_1(n+1) = j) &= \mathbb{P}(T_1(n+1) = j | T_1(n) = j)\mathbb{P}(T_1(n) = j) \\
&+ \mathbb{P}(T_1(n+1) = j | T_1(n) = j-1)\mathbb{P}(T_1(n) = j-1) \\
&= \frac{n+1-j}{n+1} \times \mathbb{P}(T_1(n) = j) \qquad \text{(apply (1) with } i = j.) \\
&+ \frac{j-1}{n+1} \times \mathbb{P}(T_1(n) = j-1) \qquad \text{(apply (1) with } i = j-1.) \\
&= \frac{n+1-j}{n+1} \times \frac{1}{n} + \frac{j-1}{n+1} \times \frac{1}{n} \qquad \text{(recall } T_1(n) \text{ is uniform)} \\
&= \frac{n}{(n+1)n} = \frac{1}{n+1},
\end{aligned}$$

which proves that $T_1(n+1)$ is uniform in $\{1, \ldots, n+1\}$.

□

This approach tells us more about the cycle decomposition of S_n. For instance it is very easy to compute the probability that i, j belong to the same cycle.

Proposition 8. *Let* $1 \leq i < j \leq n$. *Then*

$$\mathbb{P}(i, j \textit{ belong to the same cycle of } S_n) = \frac{1}{2}.$$

Proof. As all integers play the same role in S_n we have that

$$\begin{aligned}
\mathbb{P}(i, j \text{ belong to the same cycle of } S_n) &= \mathbb{P}(1 \text{ and } 2 \text{ belong to the same cycle of } S_n) \\
&= \mathbb{P}(2 \text{ does not sit at the same table as } 1) = \frac{1}{2}.
\end{aligned}$$

□

Exercise 2 Let $1 \leq i < j < k \leq n$. What is the probability that among i, j, k two of them exactly are in the same cycle?

Solution:

$$\begin{aligned}
1 &= \mathbb{P}(i, j, k \textit{ belong to the same cycle}) + \mathbb{P}(i, j, k \textit{ belong to two cycles}) + \mathbb{P}(i, j, k \textit{ belong to three cycles}) \\
&= \mathbb{P}(1, 2, 3 \textit{ belong to the same cycle}) + \mathbb{P}(1, 2, 3 \textit{ belong to two cycles}) + \mathbb{P}(1, 2, 3 \textit{ belong to three cycles}) \\
&= \frac{1}{2} \times \frac{2}{3} + \mathbb{P}(1, 2, 3 \textit{ belong to two cycles}) + \frac{1}{2} \times \frac{1}{3}
\end{aligned}$$

and finally the solution is $1 - 1/3 - 1/6 = 1/2$.

Discussion: the reinforcement phenomenon

The Chinese restaurant process illustrates the *reinforcement phenomenon* which is very common in Probability. It is also known as the "rich gets richer" phenomenon. Indeed, we observe that the more people there are at Table 1 at a given time, the more there will be in the future.

As an application, it turns out that because Table 1 appears sooner than Table 2, Table 1 is much more occupied (in average) than Table 2.

Proposition 9. *For large n, we have that*

$$\mathbb{E}[T_1(n)] \overset{n\to+\infty}{\sim} \frac{n}{2}, \qquad \mathbb{E}[T_2(n)] \overset{n\to+\infty}{\sim} \frac{n}{4}.$$

Proof. First, we claim that conditionally on the event $\{T_1(n) = i\}$, then $T_2(n)$ is uniformly distributed in $\{1, 2, \dots, n-i\}$: for every $j \leq n-i$ we have

$$\mathbb{P}(T_2(n) = j \mid T_1(n) = i) = \begin{cases} \frac{1}{n-i} & \text{if } i < n, \\ 0 & \text{if } i = n. \end{cases}$$

We skip the proof, which is very similar to the proof of Proposition 7 (in this case the combinatorial proof is easier).

Consequently, if we condition on the event $\{T_1(n) = i\}$ we have that

$$\begin{aligned} \mathbb{E}[T_2(n)|T_1(n)] &= \mathbb{E}[\text{ Uniform random var. in } \{1, 2, \dots, n - T_1(n)\}] \\ &= \frac{1 + n - T_1(n)}{2}. \end{aligned}$$

Now, by the tower property of conditional expectation[(iv)] we obtain

$$\mathbb{E}[T_2(n)] = \mathbb{E}\Big[\mathbb{E}[T_2(n)|T_1(n)]\Big] = \mathbb{E}\left[\frac{1 + n - T_1(n)}{2}\right] = \frac{1 + n - n/2}{2} \sim \frac{n}{4}.$$

□

We can go one step further and ask for the distribution of $T_3(n), T_4(n), \dots$. One can prove the following generalization of Proposition 7 (see [5, Sec.3.1], I am curious for the original reference).

Proposition 10. *Let $U_1, U_2, \dots$ be i.i.d. uniform random variables in $(0, 1)$. Then for every $k \geq 1$*

$$\left(\frac{T_1(n)}{n}, \frac{T_2(n)}{n}, \dots, \frac{T_k(n)}{n}\right) \overset{(d)}{\to} (U_1\ ,\ (1-U_1)U_2\ ,\ (1-U_1)(1-U_2)U_3\ ,\ \dots) \tag{2}$$

In particular if we consider expectations of both sides in (2) we get:

$$\left(\frac{\mathbb{E}[T_1(n)]}{n}, \frac{\mathbb{E}[T_2(n)]}{n}, \dots, \frac{\mathbb{E}[T_k(n)]}{n}\right) \overset{(d)}{\to} \left(\frac{1}{2}, \frac{1}{4}, \dots, \frac{1}{2^k}\right).$$

(iv) This says that $\mathbb{E}[\mathbb{E}[X|Y]] = \mathbb{E}[X]$.

Discussion: the size-bias phenomenon

We conclude by investigating an apparent paradox:

- In average, there are $n/2$ people at the same table as 1. But recall that the output of the Chinese restaurant process is uniform in $\mathfrak{S}_n$ so by symmetry, every element in $\{1, 2, \ldots, n\}$ plays the same role: this table can be considered as a *typical* table.
- There are in average $\log(n)$ distinct tables, so a *typical* table should have (in average) about

$$\frac{\text{Number of customers}}{\text{Number of tables}} \approx \frac{n}{\log(n)} \ll \frac{n}{2}$$

  customers.

The paradox is that Table 1 is *not* typical: by saying that 1 sits at this table the size of this table is biased. The size of Table 1 is overestimated compared to a "true" typical table. This is the *size-bias* phenomenon, whose a very nice introduction can be found in [1].

4 Applications

4.1 Applications to computer science: How to sort S_n efficiently?

We will discuss a different topic regarding random permutations: the analysis of sorting algorithms. The problem is to find an algorithm for the following problem:
Input: Sequence of numbers $x_1, x_2, \ldots, x_n$
Output: Re-ordered sequence $x_{\sigma(1)} \leq x_{\sigma(2)} \cdots \leq x_{\sigma(n)}$

We consider that the cost of the algorithm driven on $x_1, \ldots, x_n$ is given by the number of pairwise comparisons between x_i's. (We neglect in particular access to memory.)

Warm-up: the naive algorithm

As a basis for comparison we begin with a very naive algorithm.

1. Read the sequence $x_1, x_2, \ldots, x_n$ and store the minimal value $x_{\sigma(1)}$ (this requires $n-1$ comparisons),
2. Read the sequence $x_1, x_2, \ldots, x_n \setminus \{x_{\sigma(1)}\}$ and store the minimal value $x_{\sigma(2)}$ (this requires $n-12$ comparisons),
3. ...

Overall the algorithm needs

$$(n-1) + (n-2) + (n-3) + \cdots + 1 \sim \frac{n^2}{2}$$

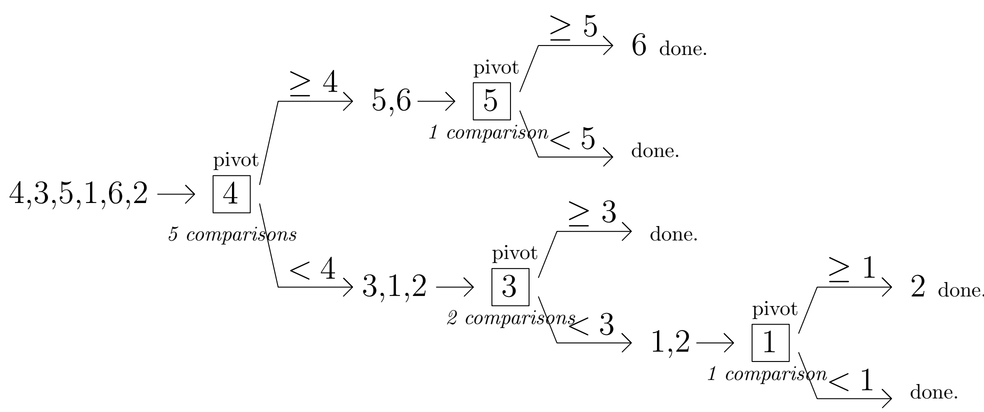
comparisons to sort the sequence. Various algorithms improve this bound and sort a list of n elements with $\mathcal{O}(n \log(n))$ comparisons. We will focus on one of them: `Quicksort`.

The algorithm `Quicksort`

The algorithm uses the *Divide-and-Conquer* strategy, there are three steps:

1. Call x_1 the *pivot* of the list.
2. Compare all the elements $x_2, \dots, x_n$ with x_1 and re-order the list so that
   (a) elements $< x_1$ come before the pivot,
   (b) elements $\geq x_1$ come after the pivot.
3. Recursively apply strategy to both sub-lists.

Here are the first steps applied to the permutation 435162:



Average-case analysis of `Quicksort`

Let $\mathrm{Comp}(x_1, \dots, x_n)$ be the number of pairwise comparisons between x_i's. For instance, in the above example we have that

$$\mathrm{Comp}(4,3,5,1,6,2) = 5 + 1 + 2 + 1 = 9.$$

If the input is random, then Comp is a random variable.

Proposition 11. *Let $S_n = (S_n(1), \dots, S_n(n))$ be a random uniform permutation of size n. Then, when $n \to +\infty$,*

$$\mathbb{E}\big[\mathrm{Comp}(S_n(1), \dots, S_n(n))\big] = 2n\log(n) + \mathrm{o}(n\log(n)).$$

Both the algorithm and its analysis were provided by Hoare [4]. A modern reference is [3].

Proof. As the execution of `Quicksort` only depends on the relative order of the elements of the sequence the continuous algorithm shows that

$$\mathrm{Comp}(S_n(1), \dots, S_n(n)) \overset{(d)}{=} \mathrm{Comp}(X_1, \dots, X_n)$$

where $X_1, \dots, X_n$ are independent random variables uniform in the interval $(0,1)$

By construction X_1 is the first pivot. Denote by $Y_1, \dots, Y_{I-1}$ be the numbers $> X_1$, and $Z_1, \dots, Z_{n-I}$, so that I is the (random) rank of X_1 in the sequence. Because of the recursive strategy the number of comparisons is given by

$$\mathrm{Comp}(X_1, \dots, X_n) = \underbrace{n-1}_{\text{Comp. with } X_1} + \mathrm{Comp}(Y_1, \dots, Y_{I-1}) + \mathrm{Comp}(Z_1, \dots, Z_{n-I}). \qquad (\star)$$

We omit the proofs of the two following claims:

- The rank I is uniform in $1, 2, \dots, n$.
- Conditionally on X_1, the Y_j's are i.i.d. (and uniform in $(0, X_1)$) and the Z_j's are i.i.d. (and uniform in $(X_1, 1)$).

Therefore, if we take expectations of both sides of $(\star)$ and put $c_n = \mathbb{E}\big[\mathrm{Comp}(X_1, \dots, X_n)\big]$ then we obtain

$$\begin{aligned} c_n &= n - 1 + \sum_{i=1}^{n} \mathbb{P}(I = i)\,(c_{i-1} + c_{n-i}) \\ &= n - 1 + \frac{1}{n}\sum_{i=1}^{n} c_{i-1} + \frac{1}{n}\sum_{i=1}^{n} c_{n-i} \\ &= n - 1 + \frac{2}{n}\sum_{i=1}^{n} c_{i-1}, \end{aligned}$$

with $c_0 = c_1 = 0$. In order to get rid of the sums we compute

$$\begin{aligned} nc_n - (n-1)c_{n-1} &= n(n-1) + 2\sum_{i=1}^{n} c_{i-1} - (n-1)(n-2) - 2\sum_{i=1}^{n-1} c_{i-1} \\ &= 2(n-1) + 2\sum_{i=1}^{n} c_{i-1} - 2\sum_{i=1}^{n-1} c_{i-1} \\ &= 2(n-1) + 2c_{n-1} \end{aligned}$$

so finally

$$nc_n = 2(n-1) + (n+1)c_{n-1}.$$

This can be rewritten as:

$$n(c_n + 2n) = 2n + (n+1)(c_{n-1} + 2(n-1)).$$

If we divide by $n(n+1)$ we get

$$\frac{c_n + 2n}{n+1} = \frac{2}{(n+1)} + \frac{c_{n-1} + 2(n-1)}{n}.$$

If we put $d_n := \frac{c_n+2n}{n+1}$ we have that

$$\begin{aligned} d_n &= \frac{2}{n+1} + \frac{2}{n} + \frac{2}{n-1} + \cdots + \frac{2}{5} + \frac{2}{4} + d_2 \\ &= \frac{2}{n+1} + \frac{2}{n} + \frac{2}{n-1} + \cdots + \frac{2}{5} + \frac{2}{4} + \frac{5}{3} \\ &= 2H_{n+1} - 2(\frac{1}{1} + \frac{1}{2} + \frac{1}{3}) + \frac{5}{3} = 2H_{n+1} - 2, \end{aligned}$$

where $H_n = \sum_{k=1}^{n} 1/k = \log(n) + \gamma + \mathrm{o}(1)$. Finally

$$c_n = 2(n+1)H_{n+1} - 2(n+1) - 2n = 2n\log(n) - 2.845569... \times n + \mathrm{o}(n).$$

□

4.2 Application to statistics: The Wilcoxon test

Imagine the following statistical situation. You want to compare two populations $\mathcal{X}$ and $\mathcal{Y}$ for which you have data $X_1, \dots, X_n$ and $Y_1, \dots, Y_m$ and specifically you want to find statistical evidences that $\mathcal{X}$ and $\mathcal{Y}$ are different.

The settings is that of a statistical test:

Hypothesis H_0:

- $X_1, \dots, X_n$ are i.i.d. with common density f (unknown)
- $Y_1, \dots, Y_m$ are i.i.d. with common density f (the same!)

Under Hypothesis H_0 we are given a sample of size $m+n$ of i.i.d. continuous random variables. That gives us a uniform permutation of size $n+m$, no matter the density f! Let us see how it allows us to design a statistical test for H_0.

For $1 \leq i \leq n$ let R_i be the rank of X_i in $\{X_1, \dots, X_n, Y_1, \dots, Y_m\}$. For $1 \leq j \leq m$ let R'_j be the rank of Y_j in $\{X_1, \dots, X_n, Y_1, \dots, Y_m\}$.

Proposition 12. *For every m, n and every i, j, if Hypothesis H_0 holds then*

$$\mathbb{E}[R_i] = \mathbb{E}[R'_j] = \frac{m+n+1}{2}.$$

In particular

$$\mathbb{E}[R_1 + \cdots + R_n] = \frac{n(m+n+1)}{2}.$$

Proof. Under H_0 random variables $R_1, \dots, R_n, R'_1, \dots, R'_m$ form a uniform permutation of size $n+m$. In particular R_i is uniform in $\{1, 2, \dots, n+m\}$ and therefore has expectation $(n+m+1)/2$. □

More generally it can be proved that for sufficiently large n, m,

$$R_1 + \cdots + R_n \approx \mathcal{N}\left(\mu_{m,n}, \sigma_{m,n}\right). \tag{3}$$

where

$$\mu_{m,n} = \frac{n(m+n+1)}{2}, \qquad \sigma_{m,n} = \frac{n(n+m)^2}{12}.$$

From this we can reject Hypothesis H_0 if $|\sum_{i \leq n} R_i - \mu_{m,n}|$ is too large. Indeed (3) can be rigorously stated as

$$\mathbb{P}\left(\left|\frac{\sum_{i \leq n} R_i - \mu_{m,n}}{\sqrt{\sigma_{m,n}}}\right| > a\right) \stackrel{n,m\to\infty}{\to} \mathbb{P}\left(|Z| > a\right)$$

where Z is a standard $\mathcal{N}(0,1)$. For $a = 1.96$ the above limit is close to 5% Thus we reject Hypothesis H_0 when

$$\left|\frac{\sum_{i \leq n} R_i - \mu_{m,n}}{\sqrt{\sigma_{m,n}}}\right| > 1.96.$$

References

[1] R.Arratia, L.Goldstein. Size bias, sampling, the waiting time paradox, and infinite divisibility: when is the increment independent? Available at https://arxiv.org/abs/1007.3910 (2010).

[2] A.D.Barbour, L.Holst, S.Janson. *Poisson approximation*. Oxford Univ. Press (1992).

[3] P.Flajolet, R.Sedgewick. *An introduction to the analysis of algorithms*. Addison-Wesley (1996).

[4] C.A.Hoare. Quicksort. *The Computer Journal*, vol.5, n.1, p.10-16 (1962).

[5] J.Pitman. *Combinatorial stochastic processes*. Lecture notes for the Saint-Flour summer school (available online) (2002).

[6] N.Pouyanne. Pólya urn models. Proceedings of *Nablus'14 CIMPA Summer School: Analysis of Random Structures*, p.65-87. Available at https://hal.archives-ouvertes.fr/hal-01214113/ (2014).

[7] Wikipedia page of the *100 prisoners problem*. https://en.wikipedia.org/wiki/100_prisoners_problem.

[8] Wikipedia page of *Rencontres numbers*. https://en.wikipedia.org/wiki/Rencontres_numbers.