

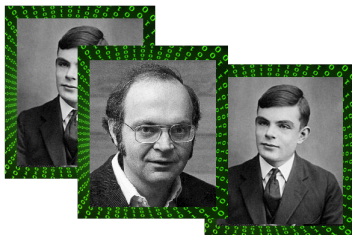
On the Biased Partial Word Collector Problem

Philippe Duchon and Cyril Nicaud

LIGM Université Paris-Est & CNRS

February 8, 2018

Coupon collector



The classical **coupon collector problem** is the following:

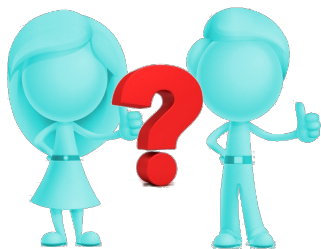
- ▶ There are n different pictures
- ▶ Each chocolate bar contains one picture, uniformly at random

Coupon collector

How many chocolate bars are required to complete the collection?

Answer: in expectation, around $n \log n$ chocolate bars are needed.

Birthday paradox



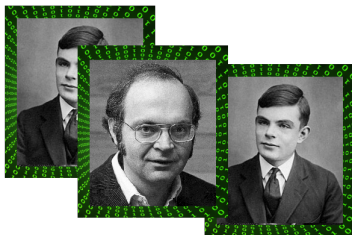
The classical **birthday paradox problem** is the following (assuming that all birthdays are uniformly random (365 possibilities)):

Birthday problem

In a room with m people, what is the probability that at least two people have the same birthday?

Answer: for $m = 23$, the probability is just above 50%

Birthday problem



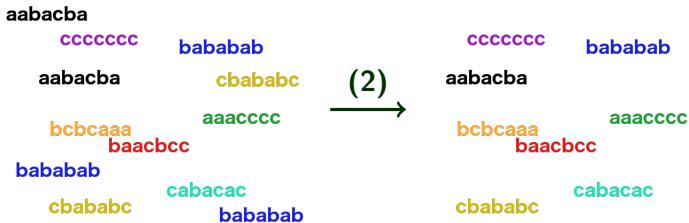
The **birthday problem** is the following:

Birthday problem

How many chocolate bars until the first duplicate?

Answer: in expectation it is $\sim \sqrt{\frac{\pi n}{2}}$

biased partial word collector problem



Our problem

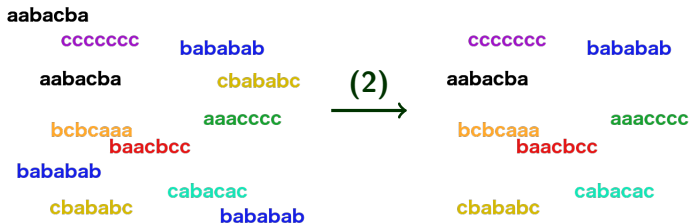
- (1) Draw N random words of length L , independently
- (2) Remove duplicates
- (3) Select a word uniformly at random

The words are generated using a **memoryless source** \mathcal{S} :

Each letter is chosen independently following a fixed probability on the alphabet. For instance $p_a = \frac{1}{3}$ and $p_b = \frac{2}{3}$ and the probability of *abba*

$$\text{is } \left(\frac{1}{3}\right)^3 \left(\frac{2}{3}\right)^2 = \frac{4}{243}$$

biased partial word collector problem



Our problem


- (1) Draw N random words of length L , independently (partial)
- (2) Remove duplicates (collector problem)
- (3) Select a word uniformly at random


The words are generated using a **memoryless source** \mathcal{S} :


Each letter is chosen independently following a fixed probability on the alphabet. For instance $p_a = \frac{1}{3}$ and $p_b = \frac{2}{3}$ and the probability of *aabba*


is $\left(\frac{1}{3}\right)^3 \left(\frac{2}{3}\right)^2 = \frac{4}{243}$ (biased)

Related works and motivations

 **Birthday paradox, coupon collectors, caching algorithms and self-organizing search.** Ph. Flajolet, D. Gardy, and L. Thimonier (DAM'92)

 **The weighted words collector.** J. Du Boisberranger, D. Gardy, and Y. Ponty (AofA'12)

 **On Correlation Polynomials and Subword Complexity.** I. Gheorghiciuc and M. D. Ward (AofA'07)

 **The number of distinct subpalindromes in random words.** M. Rubinchik and A. M. Shur. (Fundam. Inform. 16)

Subword complexity

It is the number of **distinct** factors (of a given length or not) in a string.

In our settings: $N \approx |u|$ and $L \approx |\text{factor}|$, not completely independent.

Getting started

Our problem

- (1) Draw N random words of length L , independently using \mathcal{S}
- (2) Remove duplicates
- (3) Select a word uniformly at random

Some remarks:

- ▶ There are $|A|^L$ distinct words
- ▶ If \mathcal{S} is **uniform**, then the output is a **uniform** random word
- ▶ If N is small, the output looks like a word **generated by \mathcal{S}**
- ▶ If N is large, the output looks like a **uniform** random word

By *looks like* we mean that the number of occurrences of the letters are approximatively the same.

Full statement

- ▶ The alphabet is $A = \{a_1, \dots, a_k\}$
- ▶ The probabilities for \mathcal{S} are $\mathbf{p} = (p_1, \dots, p_k)$, with $p_i = p(a_i)$
- ▶ The random variable $U_{N,L}$ denote the output of our process
- ▶ $H(\mathbf{x}) = -\sum_i x_i \log x_i$ is the classical **entropy function**
- ▶ $\text{Freq}(u) = (f_1, \dots, f_k)$ is the **frequency vector** of u with $f_i = \frac{|u|_i}{|u|}$.

Theorem [Duchon & N., LATIN'18]

Let $\ell_0 = \frac{-k}{\sum \log p_i}$ and $\ell_1 = \frac{1}{H(\mathbf{p})}$. For L sufficiently large and for any $N \geq 2$, there are three different behaviors depending on $\ell = \frac{L}{\log N}$:

- If $\ell \leq \ell_0$, then $\text{Freq}(U_{N,L}) \approx (\frac{1}{k}, \dots, \frac{1}{k})$
- If $\ell_0 \leq \ell \leq \ell_1$, then $\text{Freq}(U_{N,L}) \approx \mathbf{x}_\ell$, for some fully characterized \mathbf{x}_ℓ
- If $\ell_1 \leq \ell$, then $\text{Freq}(U_{N,L}) \approx \mathbf{p}$

$\text{Freq}(U_{N,L}) \approx \mathbf{y}$ means $\mathbb{P}(\|\text{Freq}(U_{N,L}) - \mathbf{y}\|_2 \geq \frac{\log L}{\sqrt{L}}) \leq L^{-\lambda \log L}$

Simplified statement

- ▶ The probabilities for \mathcal{S} are $\mathbf{p} = (p_1, \dots, p_k)$, with $p_i = p(a_i)$
- ▶ The random variable $U_{N,L}$ denote the output of our process
- ▶ $\text{Freq}(u) = (f_1, \dots, f_k)$ is the **frequency vector** of u with $f_i = \frac{|u|_i}{|u|}$.

Theorem [Duchon & N., LATIN'18]

There exist two thresholds $\ell_0 < \ell_1$, which depend on \mathbf{p} only, s.t. for L sufficiently large and for any $N \geq 2$, there are three different behaviors depending on $\ell = \frac{L}{\log N}$:

- If $\ell \leq \ell_0$, then $\text{Freq}(U_{N,L})$ is almost uniform
- If $\ell_0 \leq \ell \leq \ell_1$, then $\text{Freq}(U_{N,L}) \approx \mathbf{x}_\ell$, for some fully characterized \mathbf{x}_ℓ
- If $\ell_1 \leq \ell$, then $\text{Freq}(U_{N,L})$ is almost \mathbf{p}

Interpolation between the uniform distribution and \mathbf{p}

Theorem [Duchon & N., LATIN'18]

For $\ell = \frac{L}{\log N}$:

- (a) If $\ell \leq \ell_0$, then $\text{Freq}(U_{N,L})$ is almost uniform
- (b) If $\ell_0 \leq \ell \leq \ell_1$, then $\text{Freq}(U_{N,L}) \approx \mathbf{x}_\ell$, for some fully characterized \mathbf{x}_ℓ
- (c) If $\ell_1 \leq \ell$, then $\text{Freq}(U_{N,L})$ is almost \mathbf{p}

We have

$$\mathbf{x}_\ell = \left(\frac{p_1^c}{\Phi(c)}, \dots, \frac{p_k^c}{\Phi(c)} \right)$$

Where $\Phi(t) = \sum_{i=1}^k p_i^t$, and c is the unique solution in $[0, 1]$ of

$$\ell \Phi'(c) + \Phi(c) = 0.$$

Interpolation between the uniform distribution and \mathbf{p}

$$\mathbf{x}_\ell = \left(\frac{p_1^c}{\Phi(c)}, \dots, \frac{p_k^c}{\Phi(c)} \right)$$

Where $\Phi(t) = \sum_{i=1}^k p_i t$, and c is the unique solution in $[0, 1]$ of

$$\ell \Phi'(c) + \Phi(c) = 0.$$

- ▶ If $\ell = \ell_0$ then $c = 0$ and

$$\mathbf{x}_{\ell_0} = \left(\frac{p_1^0}{\Phi(0)}, \dots, \frac{p_k^0}{\Phi(0)} \right) = \left(\frac{1}{k}, \dots, \frac{1}{k} \right)$$

- ▶ If $\ell = \ell_1$ then $c = 1$ and

$$\mathbf{x}_{\ell_1} = \left(\frac{p_1^1}{\Phi(1)}, \dots, \frac{p_k^1}{\Phi(1)} \right) = \mathbf{p}$$

Proof sketch 1/3

- ▶ Let $\mathcal{W}_L(\mathbf{x})$ be the words of length L whose frequency vector is \mathbf{x}
- ▶ All the words of $\mathcal{W}_L(\mathbf{x})$ have the same probability $p(\mathbf{x})$ of being generated by the source \mathcal{S} , with $p(\mathbf{x}) = \prod p_i^{x_i L} = N^\ell \sum x_i \log p_i$
- ▶ Hence the probability $q(\mathbf{x})$ that the set contains a given word of frequency vector \mathbf{x} is $q(\mathbf{x}) = 1 - (1 - p(\mathbf{x}))^N$
- ▶ We approximate $q(\mathbf{x})$ with

$$q(\mathbf{x}) \approx \min(N p(\mathbf{x}), 1) = N^{\min(0, 1 + \ell \sum x_i \log p_i)}$$

- ▶ Since there are $\binom{L}{x_1 L, \dots, x_k L} \approx N^{\ell H(\mathbf{x})}$ words in $\mathcal{W}_L(\mathbf{x})$, the expected number of such words in the collection is roughly

$$N^{\ell \min(H(\mathbf{x}), K_\ell(\mathbf{x}))}, \text{ with } K_\ell(\mathbf{x}) = H(\mathbf{x}) + \frac{1}{\ell} + \sum x_i \log p_i$$

Proof sketch 2/3

Goal

Find the probability vector \mathbf{x} that maximises

$$\min(H(\mathbf{x}), K_\ell(\mathbf{x})), \text{ with } K_\ell(\mathbf{x}) = H(\mathbf{x}) + \frac{1}{\ell} + \sum x_i \log p_i$$

It is the minimum of two **strictly concave** functions. But we have to do some analysis in several variables x_1, \dots, x_k

Proof sketch 3/3

- ▶ For this proof sketch, let's consider that there is **only one** variable (i.e. two letters)
- ▶ Maximizing the minimum of two concave functions, two cases:



- (a)** The maximum of one function is smaller than the other function. It is the maximum of the min
 - (b)** Otherwise, the maximum is on the intersection of the two curves (which can be complicated in several dimensions)
- ▶ For our problem, the function is sufficiently nice to work with explicitly and we have
 - ▶ Case (a) appears for the two extremal ranges (uniform and **p**)
 - ▶ Case (b) appears for the middle range (interpolation), and the maximum is found using standard analysis in several variables on the hyperplane of intersection of $H(\mathbf{x})$ and $K_\ell(\mathbf{x})$

Conclusions

- ▶ There are **two thresholds**, fully characterized, for our problem
- ▶ A typical output word goes from uniformly random to distributed as an output of the source \mathcal{S}
- ▶ The **interpolation** between the two distributions is fully understood

- ▶ We focused on the distribution of letters, can we say more?
- ▶ More general sources (Markovian)?
- ▶ Distinct subpalindromes for memoryless sources?

Thanks!