Lab 7: Birthday Problem Simulator

Tommaso Massaglia
Politecnico di Torino ID: s292988
s292988@studenti.polito.it

I. Introduction

The goal of the lab is to prove the accuracy of the theoretical probability calculated in the Birthday Problem using a simulation based on an arbitrary or realistic distribution of birthdays.

II. Assumptions

The following assumptions are made:

- The birth dates of a population of size m are considered
- A birth date can be in any of n days (366 by default)
- Whenever 2 or more people share birth dates we consider it a *collision*
- Each time a group of m people is generated, we get a *boolean* output whether a collision is found or not
- We are interested in considering multiple groups of people; we define the probability of a collision for a given m as the number of experiments with a collision divided by the number of experiments ($\frac{n\,col}{n\,exp}$)
- $m < n$ always, otherwise the chance of a *collision* for that m would always be 1
- The theoretical probability for a collision to be in a group of size m is given by:

$$th\_prob \approx 1 - e^{-\frac{m^2}{2n}} \tag{1}$$

III. Input Parameters

Given the assumptions, the following input parameters are considered:

- m: the size of the population
- n: the number of days someone could be born into (366 by default)
- $n\,exp$: the number of populations of size m to run the experiments onto
- $n\,runs$: the number of times we want to test for $n\,exp$ groups of m people
- *distribution*: the distribution of birth dates

IV. Evaluation Metrics

For each m tested, we consider the *mean absolute percentage accuracy* between the obtained probability and the theoretical probability given by (1)

$$acc = 1 - \frac{1}{n\,exp \cdot n\,runs} \sum^{n\,exp} \sum^{n\,runs} \frac{|coll\_p - th\_p|}{coll\_p} \tag{2}$$

V. Data Structures and Algorithms

Regarding data structures, considering the problem is structured to check for collisions in several groups of m people I decided to define a class *population*, initialized by giving it m and a distribution, that contains the methods required to check for collisions and to run the simulation.

Most of the optimization process though was done on the algorithm part of the simulator: due to the nature of the simulation, since we are interested just in finding collisions at each step of the simulation, I decided to check for collisions and stop whenever one is found. Considering the *birthday paradox* [1] theoretical results, with an m as little as 60 we have $p \approx 0.9$ of having a collision in the population; for this reason, most of the times we have to simulate just a sixth of the available dates before finding a collision, thus making checking for collisions at each step faster than generating all birth dates and then checking for collisions ($\approx 60\%$ *faster on the PC used to simulate*).

This difference is even more evident in the extension seen in section VI-C where very large values of m are considered.

VI. Results

Three different results will be explored, one where the birth dates are uniformly distributed, one where a distribution of real-life birth dates is used to generate new birth dates, and lastly the *generalized birthday problem*.

A. Uniform Distribution

Firstly, a uniform distribution was considered to check the accuracy of the theoretical probability, the input parameters were:

seed	42
n exp	100
n runs	50
m	numpy.arange(0,100,5)
distr	uniform

The results, shown in figure 1, show how close the theoretical probability is to the simulation outcome, with an accuracy of 0.92 and close to no variance (the confidence interval for the graph was set at 0.95, if a lower number of experiments were to be run a higher variance would probably be observed).

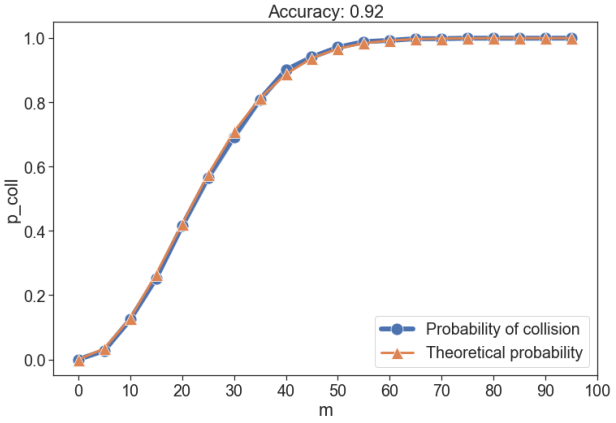

Fig. 1. m vs p_collision with a uniform distribution, ci=95%

B. Real Distribution

For this simulation, a distribution of birth dates was taken from from here, for each day a probability $\frac{n\,births}{\sum births}$ is given, thus obtaining a discrete distribution from which to generate a random birth date. The parameters for the simulation were:

seed	42
n exp	100
n runs	50
m	numpy.arange(0,100,5)
distr	external

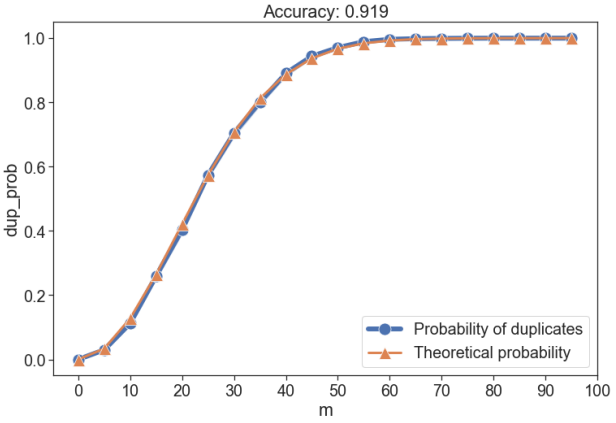

Fig. 2. m vs p_collision with a real distribution, ci=95%

Figure 2 shows how even when considering a real-life distribution of birth dates the theoretical result holds with a very high accuracy (0.92); on the contrary, the probability for a collision slightly increases as the distribution of births is uneven and it's more likely for two people to be born the same day.

C. Generalized Birthday Paradox (extension)

The *generalized birthday Paradox* asks for the minimal number $m(n)$ such that, in a set of m randomly chosen people, the probability of a collision over n days is at least 50%. For each considered m I ran *n exp* runs generating birth dates using a uniform distribution, saving at each step the number of people added to reach the first collision (and exiting the loop, thus reducing greatly the number of iterations from 10^6 to 10^3 at times); I then computed the average amount of people required to reach a collision for each considered m.

This generalization has been the subject of many studies, and as such, a number of bounds and formulas have been publicized, of which the closest one, and the one I chose as a benchmark for my simulation, is:

$$m(n) = \lceil \sqrt{2n \ln 2} + \frac{3 - 2\ln 2}{6} + \frac{9 - 4(\ln 2)^2}{72\sqrt{2n \ln 2}} - \frac{2(\ln 2)^2}{135n} \rceil \tag{3}$$

as found in [2].

The input parameters were:

seed	42
m	$[0.2, 0.4, 0.8, 1]$ · $[10^2, 10^3, 10^4, 10^5, 10^6]$
n exp	200
n runs	10

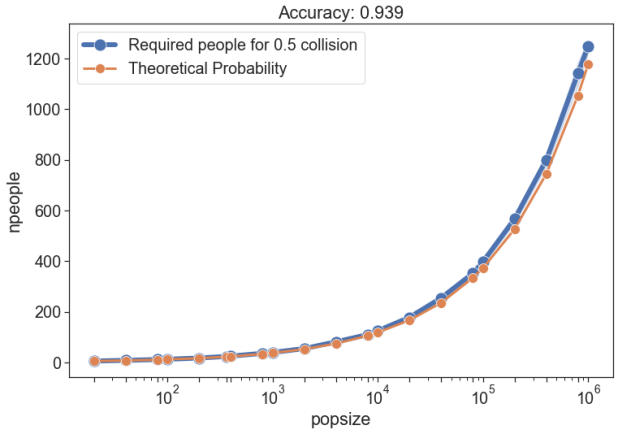

Fig. 3. number of people required to reach 50% collision vs population size in logarithmic scale, ci=95%

The graph clearly shows the high accuracy (0.939) of the theoretical bound provided in [2], which is shown to hold for all $n \leq 10^{18}$, and with very high probability for every n.

VII. Discussion

The main takeaway from these simulations is the accuracy of the theoretical probability when compared with the simulation results, which holds even when considering a real-case scenario.

Regarding the generalized birthday paradox, it's interesting to see how, despite considering larger and larger orders of magnitude, the required people to have a 50% chance of collision remains low, showing the effect of the conditional probability.

References

[1] Mario Cortina Borja; John Haigh (September 2007). "The Birthday Problem". Significance. Royal Statistical Society.

[2] D. Brink, A (probably) exact solution to the Birthday Problem, Ramanujan Journal, 2012