Lab 10 - Birthday paradox

Ali Ghasemi - s289223

Assumptions

In probability theory, the birthday problem asks for the probability that, in a set of n randomly chosen people, at least two will share a birthday. The birthday paradox is that, counterintuitively, the probability of a shared birthday exceeds 50% in a group of only 23 people. In this lab, the whole problem has been implemented both for a Uniform distribution and a distribution which is based on real-life data.

Input Parameters

Inputs of this simulation are the size of the classes that can be chosen by the user. A .CSV file that will be used for the Realistic distribution of the data, and for the optional part, the new length of the input which in the case of birthday paradox 365(days of a year)

Output Metrics

The outputs of this simulation are the confidence interval, sum of the conflicts for each distribution, plots that compare the difference between the theoretical distribution, realistic distribution, and the uniform distribution.

data structures

The main data structures used in this simulation are lists and dictionaries.

main algorithms

Different functions have been defined to achieve the distributions both for uniform

distribution and the real-life scenario.

In the function used for uniform distribution, first, we get a list of zeros a length of

366. List zeros in this list will be turned into one in case of birth on a specific day of the year. The births are generated using functions provided by python's built-in functions

The same goes for the real-life scenario but instead of random numbers, birthdays are

achieved using real-life data.

The trials (experiments) are executed 1000 times and each time, functions that are

mentioned before (to create the distributions) are called. Those functions return the

number of people that must enter the class before a collision happens.

The outputs of these two functions are stored in two different lists and after the end of the

trials, the average of these outputs is calculated (answer to task 1)

The average number of people it takes until a conflict happens is roughly equal to 23.3

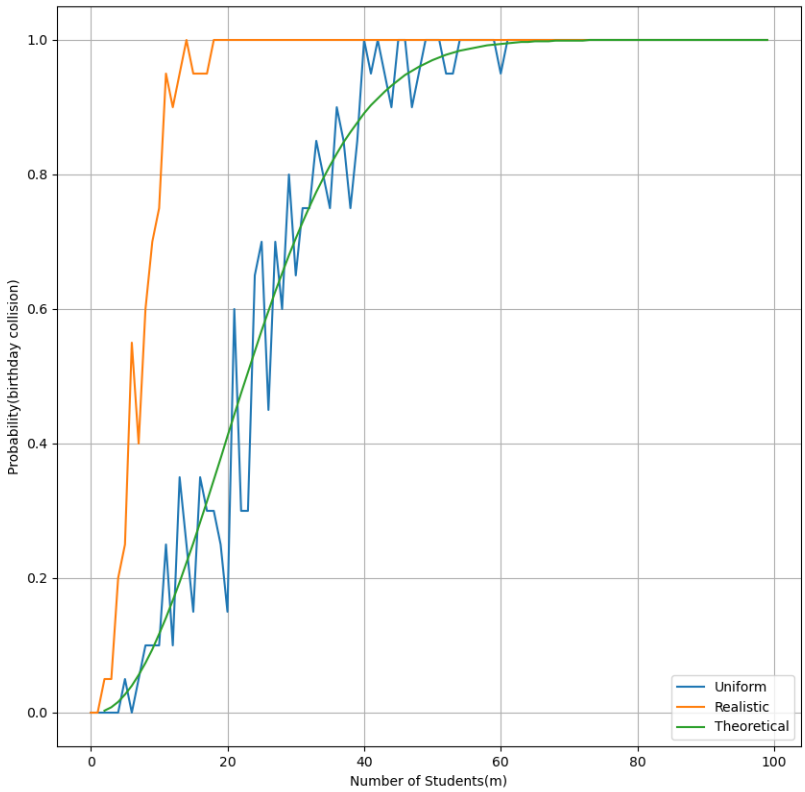
in theory and the empirical outputs that we got for the uniform distribution and real-life

data distributions are respectively equal to 24.04 and 21.042.

For the second task, the probabilities of birthday conflicts are desired and for that matter, we must create classes for students (by class, we don't mean the term "class" used in programming, we mean a normal hypothetical classroom). To perform the simulation, we consider 100 students (anything larger than this wouldn't be necessary). We also must consider that the number of students shouldn't be less than 2 (since there won't be any conflicts) and the number of trials should be higher than 32 (based on the central limit theorem).

We create the classes by making lists of zeros with lengths of 20. Again, we create random birthdays (just like task 1) and check if there are any conflicts and we count the number of conflicts and compute the probability.

You can see the comparison between the uniform distribution, real-life data distribution, and the theoretical scenario in the figure below.



To find the confidence interval, first we need to find the mean of the number of people it takes until a conflict happens both for the uniform distribution and the real-life data distribution. After that, the standard deviation must be calculated. The degree of freedom is equal to 999 (n - 1) which is the number of trials minus one. We can calculate the confidence interval using different values for confidence. In this code, 0.9 has been chosen. The obtained confidence interval for the real-life distro is equal to: (20.48960017190072, 21.594399828099284) and for the uniform distro, it is equal to: (23.60856003119988, 24.929439968800118).

Optional Part

For this part of the lab, we can try to implement the simulation with values higher than 366 for “m” to generalize the problem and use it for other scenarios.

In this code, the experiment has been implemented with m = 1000 on the uniform distribution and comparing it to the theoretical output.