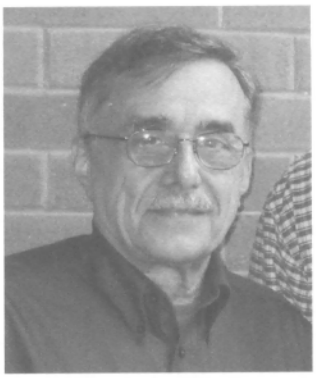
SIMULATION: Computing the Probabilities of Matching Birthdays



Bruce Trumbo

Eric Suess

Clayton Schupp

The Birthday Matching Problem

Sometimes the answers to questions about probabilities can be surprising. For example, one famous problem about matching birthdays goes like this: Suppose there are 25 people in a room. What is the probability two or more of them have the same birthday? Under fairly reasonable assumptions, the answer is greater than 50:50—about 57%.

This is an intriguing problem because some people find the correct answer to be surprisingly large. Maybe such a person is thinking, "The chance anyone in the room would have my birthday is very small," and leaps to the conclusion that matches are so rare one would hardly expect to get a match with only 25 people. This reasoning ignores that there are $(25 \times 24)/2 = 300$ pairs of people in the room that might yield a match. Alternatively, maybe he or she correctly realizes, "It would take 367 people in the room to be absolutely sure of getting a match," but then incorrectly concludes 25 is so much smaller than 367 that the probability of a match among only 25 people must be very low. Such ways of thinking about the problem are too fuzzy-minded to lead to the right answer.

As with most applied probability problems, we need to start by making some reasonable simplifying assumptions in order to get a useful solution. Let's assume the following:

- *The people in the room are randomly chosen.* Clearly, the answer would be very different if the people were attending a convention of twins or of people born in December.

Bruce Trumbo (bruce.trumbo@csueastbay.edu) is Professor of Statistics and Mathematics at California State University, East Bay (formerly CSU Hayward). He is a Fellow of ASA and holder of the ASA Founder's Award.

Eric Suess (eric.suess@csueastbay.edu), Associate Professor of Statistics at CSU East Bay, has used simulation methods in applications from geology to animal epidemiology.

Clayton Schupp, (cschupp@walk.ucdavis.edu) an MS student at CSU East Bay when this article was written, is currently a PhD student in statistics at the University of California, Davis.

- *Birthdays are uniformly distributed throughout the year.* For some species of animals, birthdays are mainly in the spring. But, for now at least, it seems reasonable to assume that humans are about as likely to be born on one day of the year as on another.

- *Ignore leap years and pretend there are only 365 possible birthdays.* If someone was born in a leap year on February 29, we simply pretend he or she doesn't exist. Admittedly, this is not very fair to those who were "leap year babies," but we hope it is not likely to change the answer to our problem by much.

The Solution Using Basic Probability

Based on these assumptions, elementary probability methods can be used to solve the birthday match problem. We can find the probability of no matches by considering the 25 people one at a time. Obviously, the first person chosen cannot produce a match. The probability that the second person is born on a different day of the year than the first is $364/365 = 1 - 1/365$. The probability that the third person avoids the birthdays of the first two is $363/365 = 1 - 2/365$, and so on to the 25th person. Thus the probability of avoiding all possible matches becomes the product of 25 probabilities:

$$P(\text{No Match}) = \prod_{i=0}^{24}\left(1 - \frac{i}{365}\right) = \frac{P_{25}^{365}}{365^{25}} = 0.4313$$

since 365^{25} is the number of possible sequences of 25 birthdays and

$$P_{25}^{365} = 25!\binom{365}{25}$$

is the number of permutations of 365 objects taken 25 at a time, where repeated objects are not permitted. Therefore,

$$P(\text{At Least 1 Match}) = 1 - P(\text{No Match}) = 1 - 0.4313 = 0.5687$$

William Feller, who first published this birthday matching problem in the days when this kind of computation was not easy, shows a way to get an approximate result using tables of logarithms. Today, statistical software can do the complex calculations easily, and even some statistical calculators can do the numerical computation accurately and with little difficulty.

```
> prod(1 - (0:24)/365)
[1] 0.4313003

> factorial(25)*choose(365, 25)/365^25
[1] 0.4313003
```

Figure 1: Two ways to calculate the probability of no matching birthdays among 25 people selected at random

In Figure 1, we show two ways to use the statistical software R to calculate the probability of no matches.

Of course, different values of n would give different probabilities of a match. With a computer package like R that has built-in procedures for doing probability computations and making graphs, it is easy to loop through various values of n and graph the relationship between n and P(At Least 1 Match). Figure 2 shows the small amount of R code required, and Figure 3 shows the resulting plot. (The labels and the reference lines were added later.)

```
p <- numeric(50)
for (n in 1:50) {
          q <- 1 - (0:(n - 1))/365
          p[n] <- 1 - prod(q) }
plot(p)
```

Figure 2: R code to calculate the probability of matching birthdays when the number of people in the room ranges from 1 to 50

By looking at the plot, we see the probability of at least one match increases from zero to near one as the number of people in the room increases from 1 to 50. We can see that n = 23 is the smallest value of n for which P(At Least 1 Match) exceeds 1/2. The computations show the probability for n = 23 to be 0.5073. A room with

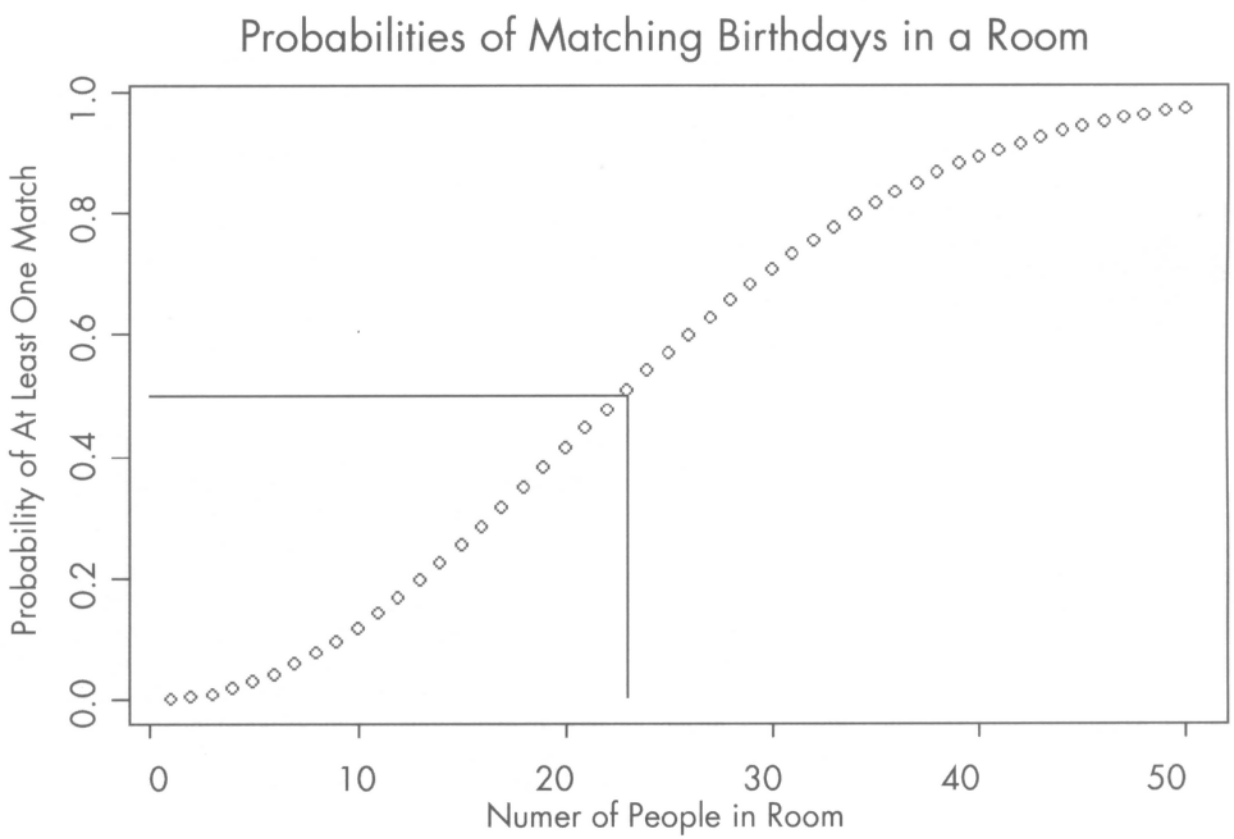


Figure 3: Plot from R of the probability of at least one pair of matching birthdays when the number of people in the room ranges from 1 to 50

n = 50 randomly chosen people is very likely to have at least one match. Indeed, for n = 50, the probability is 0.9704.

The Solution Using Simulation

A completely different approach to solving the birthday match problem is by simulation. Simulation is widely used in applied probability to solve problems that are too difficult to solve by combinatorics or other analytical methods. For example, we can use R to build a simulation model to approximate the probability that there are no matching birthdays among 25 people in a room.

This consists of first simulating the birthdays in many rooms, each with 25 people, and then checking to see what percentage of these rooms have matching birthdays. It is a little like taking a public opinion poll where the "subjects" are the rooms. We create the imaginary rooms by simulation, and then we "ask" each room, "Do you have any birthday matches?" If we ask a large number of rooms, the percentage of rooms with no match should be very near the true probability of no match in such a room.

This approach allows us to find the approximate distribution of the number of repeated birthdays (X). From this distribution, we can approximate P(X = 0), which we already know to be 0.4313. As a bonus, we also can approximate E(X), the expected number of matches among 25 birthdays. This expectation would be difficult to find without simulation methods.

Now let's build the simulation model step by step.

Step One: *Simulating birthdays for 25 people in one room*

Programmed into R is a function called "`sample`" that allows us to simulate a random sample from a finite population. To use this random sampling function, we need to specify three things.

First, we must specify the population from which to sample. For us, this is the 365 days of the year. In R, the notation `1:365` can be used to represent the list of these population elements.

Second, we have to specify how many elements of the population are to be drawn at random. Here, we want 25.

Third, we have to say whether sampling is to be done with or without replacement. Because we want to allow for the possibility of matching birthdays, our answer is "with replacement." In R, this is denoted as repl=T. We put the 25 sampled birthdays into an ordered list called b. Altogether, the R code is

```
b <- sample(1:365, 25, repl=T)
```

Each time R performs this instruction, we will get a different random list `b`. Below is the result of one run. For easy reference, the numbers in brackets give the position along the list of the first birthday in each line of output. For example, the 22nd person in this simulated room was born on the 20th day of the year, January 20.

```
 [1] 352 364 246 190 143 272 149
 [8] 206 154 272  61 199 357 141
[15] 264 157  42 340 287 166 335
[22]  20 123 214 149
```

You can see that there happen to be two matches in this list. The 6th and 10th birthdays both fall on the 272nd day of the year, and the 7th and 25th both fall on the 149th day of the year. Note that we also would have said there are two matches if, for example, the last birthday in the list had fallen on the 272nd day.

Step Two: *Finding the number of birthday matches among 25 people*

In a large-scale simulation, we need an automated way to find whether there are matching birthdays in such a room and, if so, how many repeats there are. In R, we can use the "`unique`" function to find the number of different birthdays, then subtract from 25 to find the number of birthday matches ("redundant" birthdays):

```
x <- 25 - length(unique(b))
```

For our run above, the list "`unique (b)`" is the same as `b`, but with the 10th and 25th birthdays removed. It is a list of the 23 unique birthdays since its "`length`" is 23. So the value of the random variable X for this simulated room is X = 25 – 23 = 2.

Step Three: *Using a loop to simulate X for many rooms*

If we repeat this process for a very large number of rooms, we obtain many realizations of the random variable X, and thus a good idea of the distribution of X. Counting the proportion of rooms with X = 0, we get the approximate probability of no match P(X = 0). Taking the average of these realizations of X, we get a good approximation to E(X).

When we simulated 10,000 such rooms, our result was P(No match) ≈ .4338, which is close to the exact value

0.4313 calculated using combinatorics. We also obtained $E(X) \approx 0.8081$. Additional runs of the program consistently gave values of $E(X)$ in the interval 0.81 ± 0.02.

The histogram in Figure 4 shows the approximate distribution of X – the Number of Birthday Matches. Our approximations would have been more precise if we had simulated more than 10,000 rooms, but the results seem good enough for practical purposes.

Testing Assumptions

With simulation, it is relatively easy to test the impact of the simplifying assumptions about 365 rather than 366 birthdays and that birthdays are equally likely. Actual 1997–1999 vital statistics for the United States show some variation in daily birth proportions. Monthly averages range from a low of about 94.9% of uniform in January 1999 to a high of about 107.4% in September 1999 (*www.cdc.gov/nchs/products/pubs/pubd/vsus/vsus.htm*). These fluctuations are illustrated in Figure 5. Daily birth proportions typically exceed 1/365 from May through September.

For nonuniform birthdays, computing the probability of no matches by analytical methods is beyond the scope of undergraduate mathematics; but using R, it is easy to modify our simulation so that 366 birthdays are chosen according to their true proportions in the United States population—rather than being chosen uniformly. We ran such a simulation and found that within the precision provided by 10,000 simulated rooms (about two decimal places), the results for the true proportions cannot be distinguished from the results for uniformly distributed birthdays. From these and related simulations on birthday matching, we conclude that, although birthdays in the United States are not actually uniformly distributed, it seems harmless in solving the birthday match problem to assume they are. However, important differences in the values of $P(X = 0)$ and $E(X)$ do occur if departure from uniform is a lot more extreme than in the United States population (See Nunnikhoven or Pitman and Camarri).

Using R Statistical Software

You can download R free of charge online at *www.r-project.org*. The program for doing the birthday matching problem with an explanation of the R code and an elementary tutorial on R are available online at *www.sci.csueastbay.edu/~btrumbo/bdmatch/index.html*. This website also includes further details of the birthday matching problem in a paper the authors presented at the 2004 Joint Statistical Meetings. Peter Dalgaard provides an introduction to statistics using R in *Introductory Statistics with R*. His book also is available as an electronic book, so check with your library.

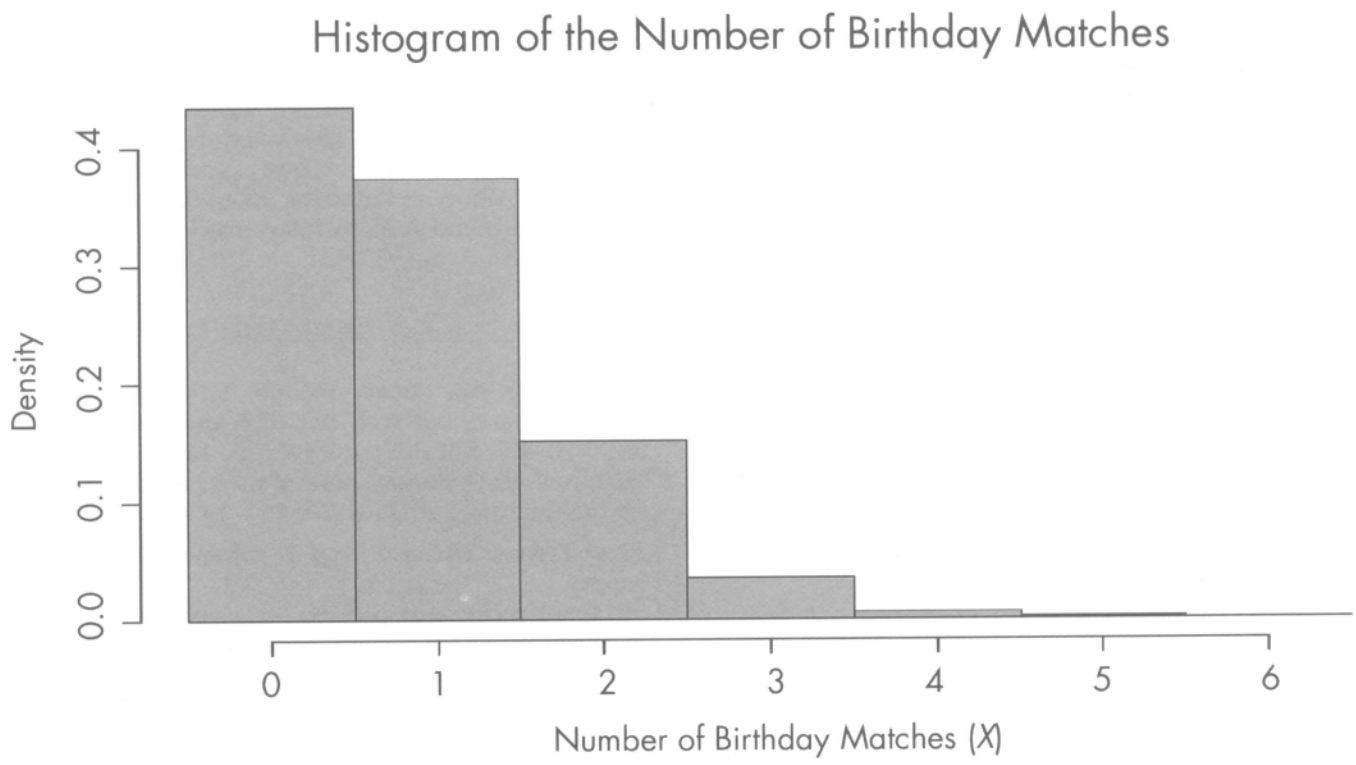


Figure 4: The simulated distribution of the number of birthday matches (X) in a room of 25 randomly chosen people

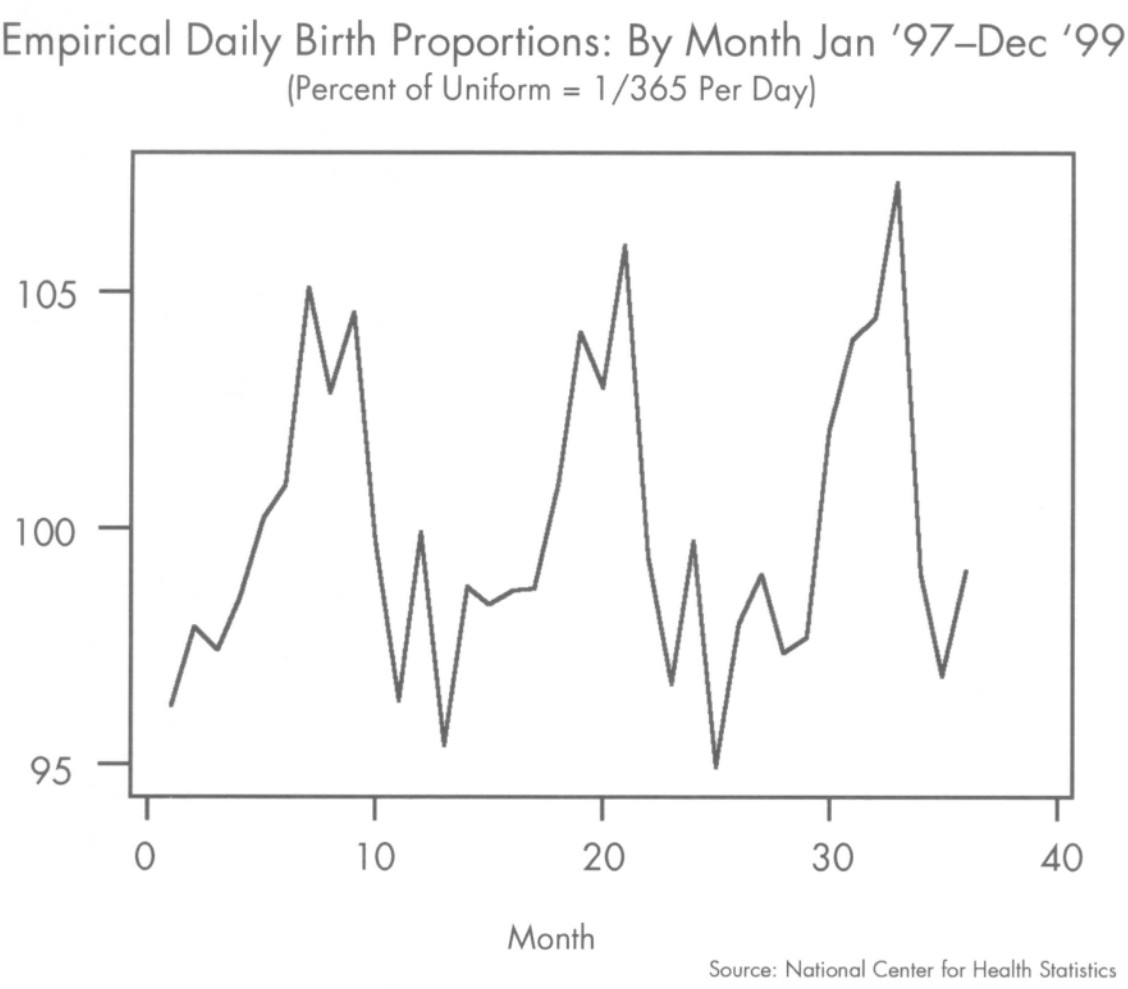


Figure 5: Cyclical pattern of birth frequencies in the United States for 36 consecutive months

Summary Comments on Simulation

Simulation is an important tool in modern applied probability modeling and in certain kinds of statistical inference. Many problems of great practical importance cannot be solved analytically.

From the birthday matching problem, we can see that—for practical purposes—simulation gives the same answer as does combinatorics for P(No Match) under the simplifying assumption that there are 365 equally likely birthdays. The same simulation provides a value for the expected number of matches, which would be difficult to find by elementary methods. Because this simulation gives what we know to be the correct answer for $P(X = 0)$, credibility is given to the value it gives for $E(X)$.

When we want to drop the uniformity assumption, we enter territory where analytic methods are much more difficult. But a minor modification of the simulation program provides us with values of $P(X = 0)$ and $E(X)$. This allows us to investigate the influence of our simplifying assumptions on results we intend to apply to real life.

In summary, we verified the correctness of a simulation method for an easy problem and then modified it to solve a closely related, but more difficult, problem. This process of building more complex simulation models based upon simpler trusted ones illustrates an important principle for using simulation reliably to solve a wide variety of important practical problems.

References

Peter Dalgaard. *Introductory Statistics with R*, Springer, 2002.

William Feller. *An Introduction to Probability Theory and Its Applications*, Vol. 1, 1950 (3rd ed.), Wiley, 1968.

Thomas S. Nunnikhoven. "A birthday problem solution for nonuniform frequencies." *The American Statistician*, 46, 270–274, 1992.

Jim Pitman and Michael Camarri. "Limit distributions and random trees derived from the birthday problem with unequal probabilities." *Electronic Journal of Probability*, 5, Paper 2, 1–18, 2000.

Bruce E. Trumbo, Eric A. Suess, and Clayton W. Schupp. "Using R to Compute Probabilities of Matching Birthdays." *2004 Proceedings of the Joint Statistical Meetings* [CD-ROM], Alexandria, Virginia: American Statistical Association. ■