First Book Using DD

Your Name

2021-09-03

Contents

Chapter 1

Hands-on excercise: the birthday problem!

As an illustration lets write a report using the participants in this workshop to illustrate the famous birthday problem.

> What is the probability that at least two people this room share the same birthday?
>
> There are 21 in this room.
>
> Is it something like $\frac{21}{365} = 0.058$?

Chapter 2

The birthday problem: the math

Actually the math says otherwise. Define p(n) as the probability that at least one pair has the same birthday, then the $1 - p(n)$ is the probability that all are born in a different day. Which we can compute as:

–

$$1 - p(n) = 1 \times \left(1 - \frac{1}{365}\right) \times \left(1 - \frac{2}{365}\right) \times \cdots \times \left(1 - \frac{n-1}{365}\right) \quad = \frac{365 \times 364 \times \cdots \times (365 - n + 1)}{365^n} = \frac{365!}{365^n (365-n)!} =$$

2.1 Code for the math

(/dynamicdocs/first_dd_solution.Rmd)

Copy and paste lines below into your `first_dd.Rmd`

```
\begin{align}
 1 -  p(n) &= 1 \times \left(1-\frac{1}{365}\right) \times
              \left(1-\frac{2}{365}\right) \times \cdots \times
              \left(1-\frac{n-1}{365}\right) \nonumber  \newline
           &= \frac{365 \times 364 \times \cdots \times (365-n+1) }{365^n} \nonumber \newline
           &= \frac{ 365! }{ 365^n (365-n)!} = \frac{n!\cdot\binom{365}{n}}{365^n}\newline
\p(n= `r n.pers`) &= `r  round(1 - factorial(n.pers) *
                         choose(365,n.pers)/ 365^n.pers, 3)`  \nonumber
\end{align}
```

Chapter 3

Don't like math? Let's run a simple simulation!

1 - Simulate 10^4 rooms with $n = 21$ random birthdays, and store the results in matrix where each row represents a room.

2 - For each room (row) compute the number of unique birthdays.

3 - Compute the average number of times a room has 21 unique birthdays, across 10^4 simulations, and report the complement.

3.1 Code for the simulation

(/dynamicdocs/first_dd_solution.Rmd)

```
birthday.prob = function(n.pers_var, n.sims_var) {
  # simulate birthdays
  birthdays = matrix(round(runif(n = n.pers_var * n.sims_var, min = 1, max = 365) ),
                     nrow = n.sims_var, ncol = n.pers_var)
  # for each room (row) get unique birthdays
  unique.birthdays = apply(birthdays, 1,
                           function(x)  length( unique(x) ) )
  # Indicator with 1 if all are unique birthdays
  all.different = 1 * (unique.birthdays==n.pers_var)
  # Compute average time all have different birthdays
  result = 1 - mean(all.different)
return(result)
}

bp_sim = birthday.prob(n.pers_var = 21, n.sims_var = 10000)
```

```
print(bp_sim)
```

```
## [1] 0.4531
```

Chapter 4

Results

- Many people originally think of a prob $\sim \frac{1}{365} \times n = 0.058$
- However the true probability is of $p(n = 21) = 0.444$
- And the simulated probability is of 0.4531]