

# MIT Introduction to Statistics 18.05 Problem Set 1

John Hancock

February 6, 2017

## Contents

<b>1</b>	<b>References and License</b>	<b>1</b>
<b>2</b>	<b>Problem 1: Poker Hands</b>	<b>2</b>
2.1	Two-Pair . . . . .	2
2.2	Three-of-a-Kind . . . . .	2
<b>3</b>	<b>Problem 2: Non-transitive Dice</b>	<b>3</b>
3.1	Probabilities and Ordering Dice . . . . .	3
3.1.1	White vs. Green and Green vs. Red . . . . .	3
3.1.2	Ordering the Dice . . . . .	4
3.2	Rolling Two Dice . . . . .	5
<b>4</b>	<b>Birthdays: counting and simulation</b>	<b>7</b>
4.1	Probability function for $\Omega$ . . . . .	7
4.2	Careful Descriptions of Subsets . . . . .	7
4.2.1	Someone Shares Our Birthday . . . . .	7
4.2.2	Two People Share a Birthday . . . . .	7
4.3	Exact Formula For Someone Sharing Our Birthday . . . . .	8
4.4	Heuristic Explanation of Previous Result . . . . .	10
4.5	R Simulation . . . . .	10
4.6	Exact Formula for $P(B)$ . . . . .	12
4.7	R Simulation to Estimate $P(C) \approx 0.5$ . . . . .	13
<b>5</b>	<b>Observed vs. Predicted Probabilities for Shared Birthdays</b>	<b>16</b>

## 1 References and License

We are answering questions in the material from MIT OpenCourseWare course 18.05, Introduction to Probability and Statistics.

Please see the references section for detailed citation information.

The material for the course is licensed under the terms at <http://ocw.mit.edu/terms>.

## 2 Problem 1: Poker Hands

### 2.1 Two-Pair

We calculate the probability of the poker dealer dealing us a hand that is a two-pair hand. First we count the number of two-pair hands, then we divide the number of two-pair hands by the total number of hands to calculate the probability of the dealer dealing us a two pair hand.

The definition of a two-pair hand is, "Two cards have one rank, two cards have another rank, and the remaining card has a third rank. e.g.  $\{2\heartsuit, 2\spadesuit, 2\clubsuit, 5\clubsuit, K\Diamond\}$ ." [8]

We take a combinations approach similar to the approach Orloff and Bloom take to calculate the probability of a one-pair hand in [12].

First we choose the ranks of the pairs. There are 13 ranks, so there are  $\binom{13}{2}$  ways to choose the ranks of the pairs.

Next we choose the suits for the cards in the pairs. There are  $\binom{4}{2}$  ways to select the suits for the cards in the first pair, and  $\binom{4}{2}$  ways to select the suits for the cards in the second pair.

To complete the hand we select one card. We have 11 ranks to choose from for the fifth card, and  $\binom{4}{1}$  ways to select its suit.

We apply the rule of product to count the number of two-pair hands:

$$\binom{13}{2} \binom{4}{2} \binom{4}{2} \binom{11}{1} \binom{4}{1} = 78 \times 6 \times 6 \times 11 \times 4 = 123552 \quad (1)$$

The number of all poker hands is the number of ways to select 5 items from a set of 52 items. Therefore the number of all poker hands is  $\binom{52}{5} = 2598960$ . Therefore the probability of a two-pair hand is  $\frac{123552}{2598960} \approx 0.048$ .

### 2.2 Three-of-a-Kind

Orloff and Bloom give the definition of a three-of-a-kind hand as, "Three cards have one rank and the remaining two cards have two other ranks. e.g.  $\{2\heartsuit, 2\spadesuit, 2\clubsuit, 5\clubsuit, K\Diamond\}$ ." [8]

We use the same approach as above.

First we select the rank for the three cards that have the same rank.

There are 13 ranks, so there are  $\binom{13}{1}$  ways to select this rank.

Next we select the suits for the three cards that have the same rank. There are 4 suits, and we choose one for each card, so there are  $\binom{4}{3}$  ways to select the suits for the 3 cards.

We have  $\binom{12}{2}$  ways to select the ranks for the fourth and fifth cards, and  $\binom{4}{1}^2$  ways to select their suits.

Now we apply the rule of product to count the number of three-of-a-kind hands:

$$\binom{13}{1} \binom{4}{3} \binom{12}{2} \binom{4}{1}^2 = 13 \times 4 \times 66 \times 16 = 54912 \quad (2)$$

Therefore the probability of a three-of-a-kind hand is  $\frac{54912}{2598960} \approx 0.021$ .

### 3 Problem 2: Non-transitive Dice

#### 3.1 Probabilities and Ordering Dice

##### 3.1.1 White vs. Green and Green vs. Red

We follow the method Orloff and Bloom use to calculate the probability that red beats white[12].

We write probability tables for white dice and green dice:

Green Die		
outcomes	1	4
probability	$\frac{1}{6}$	$\frac{5}{6}$

White Die		
outcomes	2	5
probability	$\frac{1}{2}$	$\frac{1}{2}$

Next we write the probability table for the product sample space of white and green dice:

		Green Die	
		1	4
White Die	2	$\frac{1}{12}$	$\frac{5}{12}$
	5	$\frac{1}{12}$	$\frac{5}{12}$

The pairs in the table above where the outcome for white is greater than the number for green correspond to outcomes in the product sample space where white wins. These are:  $\{white = 2, green = 1\}$ ,  $\{white = 5, green = 1\}$ , and  $\{white = 5, green = 4\}$ .

We then add the corresponding probabilities for these outcomes where white wins to calculate the probability that white wins:

$$\frac{1}{12} + \frac{1}{12} + \frac{5}{12} = \frac{7}{12} \approx 0.583 \quad (3)$$

### 3.1.2 Ordering the Dice

When we write one color beats another color, for example, "...red beats white..." we mean the outcome where the number we roll for the first color die is greater than the number we roll for the second color die. From [12], we know that there is a  $\frac{7}{12}$  probability that red beats white. From the previous subsection, we know that there is a  $\frac{7}{12}$  probability that white beats green.

Now we are required calculate the probability that green beats red.

We do the probability calculation as we do in the previous section.

We write the probability table for the red die:

Red Die		
outcomes	3	6
probability	$\frac{5}{6}$	$\frac{1}{6}$

For our convenience, we repeat the probability table for the green die:

Green Die		
outcomes	1	4
probability	$\frac{1}{6}$	$\frac{5}{6}$

Now we write the probability table for the product sample space of red and green dice:

		Green Die	
		1	4
Red Die	3	$\frac{5}{36}$	$\frac{25}{36}$
	6	$\frac{1}{36}$	$\frac{5}{36}$

The outcomes where red beats green are  $\{red = 3, green = 1\}$ ,  $\{red = 6, green = 1\}$ , and  $\{red = 6, green = 4\}$ . We sum the corresponding probabilities in the above table to compute the probability that green beats red:

$$\frac{5}{36} + \frac{1}{36} + \frac{5}{36} = \frac{11}{36} \approx 0.306 \quad (4)$$

Now we are armed with enough information to answer the question of whether or not we can order the dice from best to worst.

From [12] we know the probability that red beats white is  $\frac{7}{12}$ . We have calculated here the probability that white beats green is  $\frac{7}{12}$ , and the probability that red beats green is  $1 - \frac{11}{36} = \frac{25}{36}$ . Therefore it is more likely that red will beat white, white will beat green, and green will beat red. Therefore we cannot arrange the dice in order from best to worst.

Note: we had to look at [9] to get the answer for the ordering question.

### 3.2 Rolling Two Dice

The authors of the problem set [8] ask us, "Suppose you roll two white dice against two red dice. What is the probability that the sum of the white dice is greater than the sum of the red dice?"

The authors also suggest we watch the video [5]. The material in the video is crucial to our ability to write this answer. The text of [8] also hints that we should follow the methods in [8], and use a probability tree to answer this question.

Note: we used the example in [4] to render the probability trees below.

We start with rolling the two white dice. The nodes linked to the root of the probability tree represent the possible combinations we can roll using two white dice. The edges linking the root node of the probability tree to the nodes representing the possible combinations we can roll for the two white dice are labeled with the probability of rolling a combination of dice.

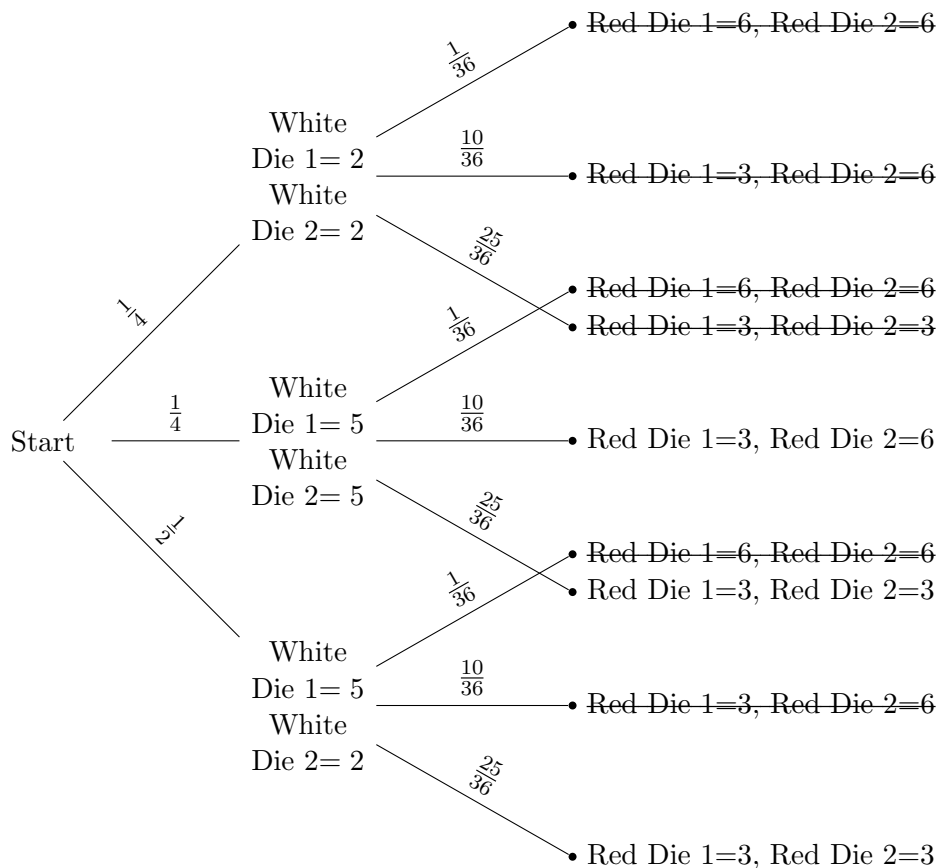
We use the multiplication rule for independent events to compute the probability of rolling a combination of white dice. The white die has three faces with a value of 5, and three faces with a value of 2, so we have a probability of  $\frac{3}{6} = \frac{1}{2}$  for rolling either a 2 or a 5. Therefore there is a probability of  $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$  of rolling two 5's or two 2's.

We use the multiplication and addition rules for independent events to compute the probability of rolling one 5 and one 2: there are two ways we can roll one 5 and one 2, so the probability of rolling one 5 and one 2 is:

$$\frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \frac{1}{2} = \frac{1}{4} + \frac{1}{4} = \frac{1}{2} \quad (5)$$

We use similar reasoning for completing the probability tree.

We cross out nodes in the tree where the path through the tree to the leaf node represents an experiment where the sum of what we roll for the white dice is not greater than the sum of what we roll for the red dice.



We use the rule of product for independent probabilities to calculate the probability of each outcome where the sum of the value we roll for the white dice is greater than the sum of the value we roll for the red dice.

We multiply the probabilities on the edges of the probability tree above where the edge is on a path to a leaf node where the label on the leaf node is not crossed out.

Path 1: With the white dice we roll two 5's, and with the red dice we roll one 3 and one 6. The probability for this outcome is  $\frac{1}{4} \times \frac{10}{36} = \frac{10}{144} \approx 0.069$ .

Path 2: We roll two 5's with the white dice, and we roll two 3's with the red dice. The probability of this outcome is:  $\frac{1}{4} \times \frac{25}{36} = \frac{25}{144} \approx 0.174$

Path 3: we roll one 5 and one 2 with the white dice, and with the red dice, we two 3's. The probability of this outcome is:  $\frac{1}{2} \times \frac{25}{36} = \frac{25}{72} \approx 0.347$

We use the addition rule for independent outcomes to compute the probability for the outcome of path 1, or path 2, or path 3.

The sum of the probabilities is approximately  $0.069 + 0.347 + 0.174 = 0.59$

## 4 Birthdays: counting and simulation

Note: we did not find this problem to be trivial, so we performed a Google search on the term, "birthday problem," to get some clues on how to proceed with a solution. We find the web pages [17], and [3] helpful in writing this solution.

### 4.1 Probability function for $\Omega$

For this problem, the elements  $\omega$  of  $\Omega$  are sequences of  $n$  birthdays, one for each person in the group.

An example  $\omega$  when  $n = 5$  is  $(1, 2, 2, 1, 1)$ .

All birthdays are equally likely.

Therefore we can write any sequence of birthdays by sampling the integers 1, 2, 3...365 with replacement.

Therefore the probability function

$$P(\Omega) = \frac{1}{365^n} \quad (6)$$

### 4.2 Careful Descriptions of Subsets

In this problem  $\Omega$  is the set

$$\{\omega \mid \omega = (x_1, x_2, \dots, x_n), x_i \in \{1, 2, 3, \dots, 365\}\} \quad (7)$$

#### 4.2.1 Someone Shares Our Birthday

Let  $S$  be the set of events where someone shares our birthday  $b$ ,  $b \in \{1, 2, \dots, 365\}$ .

Then we define  $S$  as

$$\{\omega \mid \omega = (x_1, x_2, \dots, x_n), x_i \in \{1, 2, 3, \dots, 365\}, \forall \omega (\exists x_i \mid x_i = b)\} \quad (8)$$

#### 4.2.2 Two People Share a Birthday

Let  $S$  be the set of sequences of  $n$  birthdays where at least two elements of every sequence are equal.

Then we can express  $S$  as

$$\{\omega \mid \omega = (x_1, x_2, \dots, x_n), x_i \in \{1, 2, 3, \dots, 365\}, \forall \omega (\exists x_i, x_j \mid x_i = x_j)\} \quad (9)$$

Let  $S$  be the set of sequences of  $n$  birthdays where at least three elements of every sequence are equal.

Then we can express  $S$  as

$$\{\omega \mid \omega = (x_1, x_2, \dots, x_n), x_i \in \{1, 2, 3, \dots, 365\}, \forall \omega (\exists x_i, x_j, x_k \mid x_i = x_j = x_k)\} \quad (10)$$

### 4.3 Exact Formula For Someone Sharing Our Birthday

We want to compute an exact formula for  $P(A)$  where  $A$  is the event that some sequence of birthdays  $\omega$  contains our birthday, and  $P(A)$  is the probability that a sequence of birthdays  $\omega$  contains our birthday  $b$ .

Once we have the formula, we will calculate the minimum size  $n$  of a group of people such that the probability of one of the people in the group sharing our birthday is greater than 0.5.

Note: when we write that a sequence of birthdays contains our birthday we mean that the sequence of birthdays is a member of the set we defined in the section titled, "Someone Shares Our Birthday."

There are  $365^n$  sequences of birthdays of groups of people of size  $n$  or smaller. It will be easier to calculate  $1 - P(A^c)$ .

This is because in order to calculate  $P(A)$  we would have to calculate the probability that: one person in the group shares our birthday, or two people in the group share our birthday, or three people in the group share our birthday, ..., and so on, up to the probability that all the people in the group share our birthday. Hence we would have to compute the sum of  $n$  terms.

There are  $364^n$  sequences of birthdays of groups of people of size  $n$  or smaller where none of the people in the group share our birthday. *Note:* this group is in the complement  $A^c$  of the set  $A$  of groups of people that share our birthday.

Therefore the probability of a sequence of birthdays not containing our birthday is  $\frac{364^n}{365^n}$ .

This implies that we have a formula for  $P(A)$ :

$$P(A) = 1 - \frac{364^n}{365^n} \quad (11)$$

Now we wish to find a value for  $n$  such that  $P(A) > 0.5$

Using the formula for  $P(A)$  above we can begin with:

$$1 - \frac{364^n}{365^n} > 0.5 \quad (12)$$

and solve for  $n$ .

$$1 - \frac{364^n}{365^n} > 0.5 \quad (13)$$

$$\iff \quad (14)$$



$$-\frac{364^n}{365^n} > (0.5 - 1) \quad (15)$$

$$\Longleftrightarrow \quad (16)$$

$$-\frac{364^n}{365^n} > -0.5 \quad (17)$$

$$\Longleftrightarrow \quad (18)$$

$$\frac{364^n}{365^n} < 0.5 \quad (19)$$

$$\implies \quad (20)$$

$$\log \left( \frac{364^n}{365^n} \right) < \log (0.5) \quad (21)$$

$$\Longleftrightarrow \quad (22)$$

$$\log (364^n) - \log (365^n) < \log (0.5) \quad (23)$$

$$\Longleftrightarrow \quad (24)$$

$$n \log (364) - n \log (365) < \log (0.5) \quad (25)$$

$$\Longleftrightarrow \quad (26)$$

$$n (\log (364) - \log (365)) < \log (0.5) \quad (27)$$

$$\Longleftrightarrow \quad (28)$$

$$n (\log (364) - \log (365)) < \log (0.5) \quad (29)$$

$$\Longleftrightarrow \quad (30)$$

$$n < \frac{\log (0.5)}{(\log (364) - \log (365))} \quad (31)$$

$$\Longleftrightarrow \tag{32}$$

$$n < 252.652 \tag{33}$$

Since we are calculating the probability of someone in group of size  $n$  having our birthday, we round up. So, in order for us to have more than 0.5 probability that someone in a group of people shares our birthday, there should be 253 people in the group.

#### 4.4 Heuristic Explanation of Previous Result

In the previous section, we find that there should be 253 people in a group if there is a 0.5 probability that one of the people in the group share our birthday.

If there is a sequence of numbers,  $\omega$ , that is an element of the complement of the set we defined in the section entitled, "Someone Shares Our Birthday," then the probability that one of the people in  $\omega$  shares our birthday is greater than 0.5. However, it is unlikely that in a group of  $\frac{365}{2}$  people, none of the people would share a birthday, so, we will probably need to have more than  $\frac{365}{2}$  people in the group.

#### 4.5 R Simulation

We rely on the material in [1], [10], [11], [2], [6], [14], [15], and [16] to complete this simulation.

In this section, we do an **R** simulation to estimate the smallest  $n$  for which  $P(B) > 0.9$ .

We define  $B$  as in section 9.

We define  $\Omega$  and  $\omega$  as in subsection 4.1.

Using these definitions, we write an **R** program that does the following steps to estimate values of  $P(B)$  for different values of  $n$ .

1. For  $n$  taking values 1 to 100, Generate an  $n \times m$  matrix  $B$ . Columns of  $B$  represent trials of a simulation.
  - (a)  $n$  is the number of birthdays in  $\omega$
  - (b)  $m$  is the number of trials we will do
2. Map the columns of the matrix to a list  $l1$ , where the value of an element of  $l1$  is TRUE if the number of distinct elements of a column of  $B$  is equal to the length of the column, and FALSE otherwise.
3. Create another list  $l2$  from the values of  $l1$  such that, if an element of  $l1$  is TRUE, then the corresponding element of  $l2$  is 1, if the element of  $l1$  is FALSE, then the corresponding element of  $l2$  is 0

4. Sum the number of ones in  $l_2$  to get the value  $b$ .
5. Divide  $b$  by  $m$ . The result,  $\bar{p}$ , is equivalent to an estimate of the fraction of all  $\omega \in \Omega$  that have no birthday in common.
6. Save the  $n^{\text{th}}$  result  $p = 1 - \bar{p}$ . This is the estimate of the probability that some two birthdays in  $\omega$  are the same, where  $|\omega| = n$ .

Here is a listing of the **R** program we write to carry out the algorithm outlined above:

```
# compose the length and unique functions
lengthUnique <- function(x){
  return( length(unique(x)))
}

# produce a matrix of birthdays
# column represents one trial
# number of entries in a column is number of
# people in a group
# entry is the Julian date of a person's birthday
birth_mat <- function(num_persons, num_trials){
  return ( matrix(sample(1:365, num_persons * num_trials, replace=TRUE),
    num_persons, num_trials ) )
}

# produce a matrix with n rows and 10,000 columns
#
# check if the number of unique elements in a column
# equals n, then all of the numbers in the column
# are distinct. Map columns to true or false depending
# on the outcome of this check.
#
# convert true to 1 and 0 to false
#
# sum the 1's and 0's
#
# divide by the number of columns in the matrix
#
# return the fraction of columns that have distinct
# values
simulate_no_common_birthday <- function(n){
  num_trials <- 10000;
```

```

    return (sum( as.integer( apply( x <- birth_mat(n, num_trials),
2, lengthUnique) == n ) ) / ncol(x))
}

# the fraction of columns where some two elements in the
# matrix we created when executing simulate_common_birthday
# is 1 - simulate_common_birthday
fraction_common_birthday <- function(n, min_percent){
  return ( 1 - simulate_no_common_birthday(n) )
}

# run 10 trials of finding fraction of columns where some two
# elements of a column are the same, where the length of a column
# goes from 1 to 100.
# This simulates finding the fraction of groups of people where
# some two people in the group share a birthday, for groups ranging
# in size from 1 to 100.
# We then return the position of the first element in the list greater
# than the minimum percentage.
birthday_experiment <- function(min_percentage){
  return( min(which( apply(matrix(c(1:100), 1, 100), 2, fraction_common_b
}

# do the birthday experiment a few times
print(apply(matrix(rep(0.9, 4), 1, 4), 2, birthday_experiment));

```

The output of running this program is:

```
[1] 41 42 41 41
```

We see that three out of four trials give an estimate that  $|\omega|$  estimate of  $n = 41$  for  $P(B) > 0.9$ , and one trial gives an estimate of  $n = 42$ . Therefore we estimate that for  $P(B) > 0.9$ , the smallest  $n$  possible is  $n = 41$ .

We run the simulation again for 30 trials instead of 10,000 trials.

We make a small modification to the program listed above to accomplish this.

The output of the modified program is:

```
[1] 38 33 34 41
```

The estimates for  $n$  are more variable, therefore the So the estimated probabilities are also more variable when we do a smaller number of trials.

#### 4.6 Exact Formula for $P(B)$

An element of  $\omega \in \Omega$  is a sequence of birthdays.

We gave a formal definition of  $\Omega$  in 4.2

A birthday is an integer from 1 through 365. All birthdays are equally likely, and two or more people can be born on the same day, so any  $\omega$  of length  $n$  is a sample with replacement from the set  $\{1, 2, 3, \dots, 365\}$ . We will use  $365^n$  for the number of  $\omega \in \Omega$  for a given  $n$  is.

Using  $365^n$  for the total number of  $\omega \in \Omega$  means that the order of birthdays in a given  $\omega$  is important. To see this, consider that for  $n = 2$  two elements of  $\Omega$  are  $(5, 6)$  and  $(6, 5)$ .

It will be easier to calculate the probability  $\bar{P}$  that for an  $\omega$  containing  $n$  birthdays, none of the birthdays are the same. Then the probability  $P$  that some two birthdays in  $\omega$  are the same is  $1 - \bar{P}$ .

It is easier to calculate  $1 - \bar{P}$  because in order to calculate  $P$  directly, we have to take into account that there are  $\binom{n}{2}$  ways to select two birthdays in  $\omega$  to be the same, or  $\binom{n}{3}$  ways to select three birthdays in  $\omega$  to be the same, or so on.

To calculate  $1 - \bar{P}$  we must count the number of ways to select  $n$  birthdays from a set of 365 birthdays where no two birthdays are the same.

Recall that we are using  $365^n$  as the total number of  $\omega \in \Omega$ , and the order of birthdays in a given  $\omega$  is important to us.

The number of samples of size  $n$  of 365 elements, where the order is important is  $365P_n$ . These samples of size  $n$  meet the definition of elements of  $\Omega$ .

Therefore the probability  $\bar{P}$  that some  $\omega \in \Omega$  has  $n$  distinct birthdays is:

$$\bar{P} = \frac{365P_n}{365^n} \quad (34)$$

Therefore the exact formula for  $P(B)$  is

$$P(B) = 1 - \frac{365P_n}{365^n} \quad (35)$$

## 4.7 R Simulation to Estimate $P(C)$ ; 0.5

Note: we rely on material in [13] to write the **R** simulation in this section. Our approach for this simulation is to generate a sequence of birthdays using the sample function, sort the elements of the sample, and check if any three consecutive elements are all the same.

We repeat this approach for 10,000 trials.

Here is a listing of the **R** Simulation:

```
# compose the length and unique functions
lengthUnique <- function(x){
  return( length(unique(x)))
}
```

```
# return an array of 3-combinations of x
```

```

threeCombn <- function(x) {
  return (combn(x, 3))
}

# generate a matrix M where columns of the matrix represent a
# list of birthdays
birth_mat <- function(num_persons, num_trials){
  return ( matrix(sample(1:365, num_persons * num_trials, replace=TRUE),
    num_persons, num_trials ) )
}

# for the number of trials
# produce a sample of length n
# sort the sample
# check if 3 consecutive elements of the sorted sample are the same
# break to avoid overcounting in the case
# where more than 3 people in the group share
# the same birthday
simulate_3_common_birthday <- function(n){
  num_trials <- 10000;
  num_three_same = 0;
  for (i in 1:num_trials){
    x = sample(1:365, n, replace=TRUE);
    x = sort(x);
    for (j in 1:(length(x)-2)){
      if (x[j] == x[j+1] && x[j+1] == x[j+2]){
        num_three_same = num_three_same + 1;

        break;
      }
    }
  }
  return(num_three_same/num_trials);
}

# run 10 trials of finding fraction of columns where some three
# elements of a column are the same, where the length of a column
# goes from 1 to 100.
# This estimates the perentage of groups of people where
# some three people in the group share a birthday, for groups ranging
# in size from 1 to 100.
# We then return the postion of the first element in the list greater
# than the minimum percentage.

```

```

birthday_experiment <- function(min_percentage){
  return(  apply(matrix(c(86:88), 1, 3), 2, simulate_3_common_birthday)
)
}

```

```

# do the birthday experiment a few times
  birthday_experiment(0.5)
#simulate_3_common_birthday(78)

```

We ran this program 3 times to produce the following outputs:

```

[1] 0.4878 0.5072 0.5112
[1] 0.4877 0.4996 0.5082
[1] 0.4882 0.4980 0.5066

```

Therefore we find that the minimum value for  $n$  for which  $P(C)$  is 88.

## References

- [1] Stack Overflow User 42. *R: Test if all elements of a list (lists themselves) are equal (Answer)*. Available at <http://stackoverflow.com/questions/20597547/r-test-if-all-elements-of-a-list-lists-themselves-are-equal> (2013/12/15).
- [2] Stack Overflow User dsliberty. *Convert Dataframe Column to 1 or 0 for true/false Values and Assign to dataframe, Answer from dsliberty*. Available at <http://stackoverflow.com/questions/33930188/convert-dataframe-column-to-1-or-0-for-true-false-values-and-assign-to-data> (2015/11/26).
- [3] Better Explained. *Birthday Problem*. Available at <https://betterexplained.com/articles/understanding-the-birthday-paradox/> (2007/12/31).
- [4] Kjell Magne Fauske. *Example: Probability tree*. Available at <http://www.texample.net/tikz/examples/probability-tree/> (2006/12/14).
- [5] James Grime and David Spiegelhalter. *Non-transitive Dice*. Available at [http://www.youtube.com/watch?v=zWUrwhaqq\\_c](http://www.youtube.com/watch?v=zWUrwhaqq_c) (2010/08/05).
- [6] Robert L. Kabacoff. *User Written Funcions*. Available at <http://www.statmethods.net/management/userfunctions.html> (2017).
- [7] Chris Mulligan. *Births By Day of Year*. <http://chmullig.com/2012/06/births-by-day-of-year/> (2015).
- [8] Jeremy Orloff and Jonathan Bloom. *18.05 Problem Set 1, Spring 2014*. Available at [https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/assignments/MIT18\\_05S14\\_ps1.pdf](https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/assignments/MIT18_05S14_ps1.pdf) (Spring 2014).

- [9] Jeremy Orloff and Jonathan Bloom. *18.05 Problem Set 1, Spring 2014 Solutions*. Available at [https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/assignments/MIT18\\_05S14\\_ps1\\_solutions.pdf](https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/assignments/MIT18_05S14_ps1_solutions.pdf) (Spring 2014).
- [10] Jeremy Orloff and Jonathan Bloom. *18.05 R Tutorial 1A: Basics*. Available at <https://ocw.mit.edu/ans7870/18/18.05/s14/html/r-tut1a.html> (Spring 2014).
- [11] Jeremy Orloff and Jonathan Bloom. *18.05 R Tutorial 1B: Random Numbers*. Available at <https://ocw.mit.edu/ans7870/18/18.05/s14/html/r-tut1b.html> (Spring 2014).
- [12] Jeremy Orloff and Jonathan Bloom. *Probability: Terminology and Examples 18.05 Spring 2014*. Available at [https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/class-slides/MIT18\\_05S14\\_class2slides.pdf](https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/class-slides/MIT18_05S14_class2slides.pdf) (Spring 2014).
- [13] R Core Team and contributors worldwide. *Logical Operators*. Available at <https://stat.ethz.ch/R-manual/R-devel/library/base/html/Logic.html> (2017/2/3).
- [14] R Core Team and contributors worldwide. *Replicate Elements of Vectors and Lists*. Available at <https://stat.ethz.ch/R-manual/R-devel/library/base/html/rep.html> (2017/2/3).
- [15] R Core Team and contributors worldwide. *The Number of Rows/-Columns of an Array*. Available at <https://stat.ethz.ch/R-manual/R-devel/library/base/html/nrow.html> (2017/2/3).
- [16] Andrie de Vries and Joris Meys. *How to Apply Functions on Rows and Columns in R*. Available at <http://www.dummies.com/programming/r/how-to-apply-functions-on-rows-and-columns-in-r/> (Available 2016/2/4).
- [17] Wikipedia. *Birthday Problem*. Available at [https://en.wikipedia.org/wiki/Birthday\\_problem](https://en.wikipedia.org/wiki/Birthday_problem) (2017/01/23).

## 5 Observed vs. Predicted Probabilities for Shared Birthdays

In [7], we see that the observed value for the probability of people sharing birthdays is higher than the value we predict with an exact formula.

We posit that this is so because when a birth involves a medical procedure such as induced labor women schedule the birth date at convenient times.