Probability Fundamentals

Blake Pappas

9/21/2021

Simulation

In probability and statistics, direct calculation can become difficult for even moderately complicated probabilities. In these situations, simulation can be used to find approximate probabilities.

This strategy involves writing a program that simulates the probability experiment. Repeat the experiment many times, say, several thousand times. For each repetition of the experiment, keep track of whether the event E occurred. The approximate probability of event E is the number of times E occurred divided by the number of simulations.

A probability experiment that results in one of two possible outcomes is sometimes called a Bernoulli trial. A coin flip is an example of a Bernoulli trial with which we are all familiar. Here are three ways to simulate such an experiment using R.

1. Use the `sample` function to generate random integers. The first argument in `sample` is the sample space, and the `prob =` argument can be used to specify the probabilities of each outcome. If `prob =` is not used, each outcome is assumed to have equal probability.

```
sample(c("Heads", "Tails"), size = 1, replace = TRUE)  # Use 1 and 2 to correspond to heads and tails

## [1] "Tails"

sample(c("Heads", "Tails"), size = 1, replace = TRUE, prob = c(0.75, 0.25)) # Use the prob option to use an unfair coin.

## [1] "Heads"

# You Can Also Use Integers to Represent the Outcomes
sample(c(450, 100), size = 1, replace = TRUE)

## [1] 450
```

2. Use `rbinom` to sample a 0 or 1, where the option `prob =` specifies the probability of getting a 1.

Here is R code that will simulate coin flips using this function.

```
rbinom(1, size = 1, prob = 0.5) # Flip a fair coin one time

## [1] 1
```

```r
rbinom(1, size = 1, prob = 0.75) # Flip a coin that favors the "1" outcome one time

## [1] 1

rbinom(20, size = 1, prob = 0.5)  # Flip a fair coin twenty times

##  [1] 0 1 0 1 0 1 0 1 1 1 1 1 0 1 0 0 1 0 0 1
```

3. Sample a random number between 0 and 1. The `runif` function uses a distribution for which all numbers between 0 and 1 are equally likely. Then, consider the draw a head if it is less than 0.5 and tails otherwise (or vice versa).

```r
u <- runif(1)
if(u < 0.5) {
  outcome <- 'heads'} else {
  outcome <- 'tails'}
outcome

## [1] "heads"
```

(Note: option 3 might seem a bit roundabout, but uniform numbers are at the heart of many simulation strategies. It's a good strategy to be familiar with.)

Example of Using Simulation: Cards

Consider an experiment in which five cards are dealt at random from a deck of 52. The goal is to compute the probability of event E, defined as "getting exactly one Ace" in the hand.

Combinatoric Approach

To find this probability exactly, we need to do some counting. The P(E) will be

$$P(E) = \frac{\text{number of outcomes in E}}{\text{number of possible outcomes}}.$$

First, how many possible outcomes are there in the experiment? In other words, how many unique hands of five cards are possible? Order does not matter, so we can use the binomial coefficient to count the number of ways to choose 5 items from 52.

$$\binom{52}{5} = \frac{52!}{47!\,5!}$$

That number is the denominator of the probability.

Next, how many outcomes are in E? In other words, how many possible hands include exactly one ace? This is a bit trickier. There are four ways to select the one ace out of four aces in the deck (more precisely, $\binom{4}{1}$). Then, for the other four cards, there are 48 possibilities remaining. The number of hands is then

$$\binom{4}{1} \cdot \binom{48}{4}.$$

The probability of getting exactly one ace, then, is

$$\frac{\binom{4}{1} \cdot \binom{48}{4}}{\binom{52}{5}},$$

or numerically, 0.2995.

Here is R code to compute the value:

```
choose(4, 1) * choose(48, 4) / choose(52, 5)

## [1] 0.2994736
```

Simulation Approach

First, we create a vector called ‘deck’, which represents the 52 cards that might be chosen. We could have included the suit, but did not for this problem since we are just counting aces.

```
# Create a Vector Called "Deck" Which Represents the 52 Cards
deck <- rep(c(2:10, 'J', 'Q', 'K', 'A'), 4)
deck

##  [1] "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "J"  "Q"  "K"  "A"  "2" 
"3" 
## [16] "4"  "5"  "6"  "7"  "8"  "9"  "10" "J"  "Q"  "K"  "A"  "2"  "3"  "4" 
"5" 
## [31] "6"  "7"  "8"  "9"  "10" "J"  "Q"  "K"  "A"  "2"  "3"  "4"  "5"  "6" 
"7" 
## [46] "8"  "9"  "10" "J"  "Q"  "K"  "A"
```

To simulate drawing a hand of 5 cards without replacement, use the sample function with the `replace = FALSE` option.

```
hand <- sample(deck, 5, replace = FALSE)
hand

## [1] "6"  "2"  "10" "J"  "6"
```

To simulate the probability, we will repeat this experiment 10,000 times. For each repetition of the experiment, we will track whether or not the hand contained exactly one ace.

```
n_simulations <- 1e4  # Number of times to simulate the experiment
one_ace <- rep(NA, n_simulations)  # Will contain indicators of whether the 
hand had one Ace.

for(i in 1:n_simulations)
```

```
{
  # Deal a hand from the deck
  hand <- sample(deck, 5, replace = FALSE)

  if(sum(hand == 'A') == 1)  # Check if there is exactly on "A" in hand
  {
    one_ace[i] <- 1
  }
  else
  {
    one_ace[i] <-0
  }
}
```

Now, the object one_ace is a vector of length 10,000 elements, each corresponding to one simulation of the experiment. The 1's indicate the times when we got exactly one ace. To approximate the probability of exactly one ace, we just calculate how many 1's are in the one_ace vector.

```
sum(one_ace == 1) / n_simulations

## [1] 0.2955
```

Example: Dice and Independence

Two fair dice are rolled.

a. Write the sample space for this experiment. Give each outcome as an ordered pair; e.g., (4, 2) indicates that die 1 is a four and die 2 is a 2.

```
  #S = {(1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1),
  # (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (2, 6),
  # (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (3, 6),
  # (4, 1), (4, 2), (4, 3), (4, 4), (4, 5), (4, 6),
  # (5, 1), (5, 2), (5, 3), (5, 4), (5, 5), (5, 6),
  # (6, 1), (6, 2), (6, 3), (6, 4), (6, 5), (6, 6)}
```

b. Find the probabilities of the following events:

Event A: Exactly one dot appears on each of the two upturned faces.

Event B: The sum of the dots on the two upturned faces is exactly 4.

Event C: The sum of the dots on the two upturned faces is at most 4.

Event D: The sum of the dots on the two upturned faces is equal to 6.

Event E: The same number of dots appears on both upturned faces.

Answer: P(A) = 0.0278; P(B) = 0.0833; P(C) = 0.1667; P(D) = 0.1389; P(E) = 0.1667

c. Find the following conditional probabilities:

P(A|C)

P(B|C)

P(D|E)

P(A|B)

Answer: P(A|C) = 0.1667; P(B|C) = 0.5; P(D|E) = 0.1667; P(A|B) = 0.3333

d. Are events A and C independent? Use a probability calculation to explain your answer.

Answer: Events A and C are not independent because P(A|C) <> P(A) and P(A∩C) <> P(A) * P(C). For the first formula, P(A|C) = 0.1667 while P(A) = 0.0278. For the second formula, P(A∩C) = 0.1667 and P(A) * P(C) = 0.0046.

e. Are events A and C disjoint? Explain your answer.

Answer: Events A and C are not disjoint because P(A∩C) <> ▯. P(A∩C) is equal to 0.1667 which does not equivalent to ▯, or an empty sample space.

Example: Simulating Probabilities

An economics exam has 25 questions. Each question is True/False and the student passes if they answer 17 or more questions correctly. Consider a student who flips a fair coin and answers “true” if the coin is a heads and “false” if the coin is a tails.

a. What is the probability that the student will answer exactly 12 questions correctly using this strategy?

Answer: There is a 0.48 probability that the student will answer exactly 12 questions correctly.

b. Write a simulation to approximate the probability that the student will pass the exam.

Answer: Please see below for the simulation. The value of `prob` represents the probability that the student will pass the exam.

```
quarter <- rep(c('Heads', 'Tails'))
quarter

## [1] "Heads" "Tails"

coin <- sample(quarter, 1, replace = FALSE)
coin

## [1] "Tails"

n_simulations <- 10000
seventeen_correct <- rep(NA, n_simulations)
```

```
for(i in 1:n_simulations)
{
  # Deal a hand from the deck
  coin <- sample(quarter, 1, replace = FALSE)

  if(sum(coin == 'Heads') == 1)  # Check if there is exactly on "A" in hand
  {
    seventeen_correct[i] <- 1
  }
  else
  {
    seventeen_correct[i] <-0
  }
}

prob <- sum(seventeen_correct == 1) / n_simulations
prob

## [1] 0.4998
```

Example: Phone Calls

My telephone rings 12 times each week, the calls being randomly distributed among the seven days.

What is the probability that I get at least one call each day? Use combinatorics to arrive at an exact answer or simulation to approximate the exact answer.

Answer: The probability of getting at least one call each day is 0.1114. I calculated the probability using the following logic, assuming no replacement:

$$\frac{12 \cdot 11 \cdot 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6}{12 \cdot 12 \cdot 12 \cdot 12 \cdot 12 \cdot 12 \cdot 12}$$

Example: Utility Payments

The utility company in a large metropolitan area finds that 85% of its customers pay a given monthly bill in full.

Four customers are chosen at random from the list of all customers. Find the probability that none of the four customers will pay the monthly bill in full.

Answer: The probability that none of the four customers will pay the monthly bill in full is 0.0005. If the probability of its customers paying a given monthly bill in full is 85%, then probability of them not paying it in full is 15%. 0.15^4 = 0.0005

b. One customer is selected at random. We want to find the probability that the customer pays the bill in full in the next two consecutive months. Explain why finding 0.85^2 is not the correct approach.

Answer: Finding 0.85^2 is not the correct approach because the two events are independent of one another. Knowing the outcome for one given month does not affect the probability of that same event occurring in a different month.

Example: Draft Lottery

In a draft lottery containing the 366 days of the year (including February 29), what is the probability that 180 days, randomly drawn without replacement, contain no days in September?

Use combinatorics to find the probability.

Answer: The probability that 180 days, randomly drawn without replacement, contain no days in September is 5.398333e-78. I calculated this using the following logic:

$$\frac{\binom{150}{30}}{\binom{366}{180}}$$

```
choose(150, 30) / choose(366, 180)

## [1] 5.398333e-78
```

Example: Disjointness

If $P(A) = \frac{1}{3}$ and $P(B') = \frac{1}{4}$, can A and B be disjoint? Explain.

Answer: No, A and B cannot be disjoint. If this were the case, then P(A) □ P(B) > 1 because 1/3 + 3/4 = 13/12. In probability, an outcome can only be between 0 and 1.