Teaching demo: Intro to simulation

Ciaran Evans

Context

- This lecture has been used as the first lecture in STA 279 Statistical Computing

- Students in Statistical Computing have taken STA 112 and are familiar with some R fundamentals; no other background is assumed

- First unit of Statistical Computing is on simulation studies
    - Allows review of some R basics
    - Provides context and motivation for fundamental computing concepts: data types, iteration, good coding practices

Today's lesson

- **Learning goal:** by the end of this lesson, students will be able to implement a short simulation to answer a probability question
- Topics reviewed or introduced:
  - Planning simulations
  - Vectors in R
  - Iteration
- Student participation:
  - Short neighbor/group discussions
  - Dialogue and questions throughout
  - Your turn: activity at the end of the lesson

Warm-up question

Problem: 10 people are at a party, and all of them are wearing hats. They each place their hat in a pile; when they leave, they choose a hat at random. What is the probability at least one person selected the correct hat?

Question: Work with your neighbor to discuss the following question:

- Without calculating probabilities, how could you design an experiment to estimate this probability?

Designing an experiment

Step 1: representing the hats

```r
hats <- 1:10

hats
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```r
hats[3]
```

```
## [1] 3
```

- `hats` is a **vector**, containing the numbers 1 to 10
- entries in a vector are accessed by their index

Step 2: everyone draws a random hat

```r
hats <- 1:10
randomized_hats <- sample(hats, size = 10,
                          replace = FALSE)

hats
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```r
randomized_hats
```

```
##  [1] 10  3  7  4  2  6  5  9  8  1
```

- The `sample` function creates a random sample from a vector
- How many people selected their original hat?

Step 3: check who got their original hat

```
hats
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
randomized_hats
```

```
##  [1] 10  3  7  4  2  6  5  9  8  1
```

Step 3: check who got their original hat

```
hats
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
randomized_hats
```

```
##  [1] 10  3  7  4  2  6  5  9  8  1
```

```
hats == randomized_hats
```

```
FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
```

Step 3: check who got their original hat

```
hats
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
randomized_hats
```

```
##  [1] 10  3  7  4  2  6  5  9  8  1
```

```
hats == randomized_hats
```

```
FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
```

```
# TRUE is 1, FALSE is 0
sum(hats == randomized_hats)
```

```
## [1] 2
```

Step 3: check who got their original hat

```
hats
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
randomized_hats
```

```
##  [1] 10  3  7  4  2  6  5  9  8  1
```

```
hats == randomized_hats
```

```
FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
```

```
# TRUE is 1, FALSE is 0
sum(hats == randomized_hats)
```

```
## [1] 2
```

```
# did at least one person get their hat?
sum(hats == randomized_hats) > 0
```

```
## [1] TRUE
```

Code so far

```r
hats <- 1:10
randomized_hats <- sample(hats, size = 10,
                          replace = FALSE)
```

```r
sum(hats == randomized_hats) > 0
```

```
## [1] TRUE
```

- In this case, at least one person received their original hat!
- Is this a good estimate of the *probability*?

Step 4: iteration

A `for` loop repeats code many times:

```r
nsim <- 10000 # number of simulations
for(i in 1:nsim){


}
```

Loop example

```r
for(i in 1:5){
  print(3)
}
```

```
## [1] 3
## [1] 3
## [1] 3
## [1] 3
## [1] 3
```

Loop example

```r
for(i in 1:5){
  print(i)
}
```

Loop example

```r
for(i in 1:5){
  print(i)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
```

Step 4: iteration

A `for` loop repeats code many times:

```
nsim <- 10000 # number of simulations
hats <- 1:10

for(i in 1:nsim){
  randomized_hats <- sample(hats, size = 10,
                            replace = FALSE)
  print(sum(hats == randomized_hats) > 0)
}
```

```
## [1] TRUE
## [1] TRUE
## [1] TRUE
## [1] TRUE
## [1] TRUE
## [1] TRUE
## [1] FALSE
## [1] FALSE
```

Step 4: iteration

A `for` loop repeats code many times:

```r
nsim <- 10000 # number of simulations
hats <- 1:10
results <- rep(NA, nsim) # vector to store results

for(i in 1:nsim){
  randomized_hats <- sample(hats, size = 10,
                            replace = FALSE)
  results[i] <- sum(hats == randomized_hats) > 0
}

head(results)
```

```
## [1]  TRUE  TRUE FALSE FALSE FALSE  TRUE
```

Step 4: iteration

A `for` loop repeats code many times:

```r
nsim <- 10000 # number of simulations
hats <- 1:10
results <- rep(NA, nsim) # vector to store results

for(i in 1:nsim){
  randomized_hats <- sample(hats, size = 10,
                            replace = FALSE)
  results[i] <- sum(hats == randomized_hats) > 0
}

mean(results)
```

```
## [1] 0.6307
```

Class activity

For the remainder of class, work with a neighbor on the class activity (link below and on the course website):

https://sta279-example.github.io/class_activities/ca_lecture_1.html

What comes next?

- Continuing probability simulations (gambler's ruin, airplane seating, Monty Hall problem, etc.)
    - setting seeds
    - good coding practices
    - `for` and `while` loops
    - nested loops
    - `if...else if...else` statements

- Statistical simulations
    - answering questions about linear regression models (e.g., does constant variance matter?)
    - ADEMP framework[1]
    - introduction to writing functions

[1] "Using simulation studies to evaluate statistical methods" (Morris *et al.* 2019)

What is the exact probability?