Lecture 2

Agenda

- HW 1 released on course website, due next Friday
- Continuing intro to probability simulations
- Time permitting: start HW 1 in class
- Office hours (Manchester 329):
  - 3 - 4pm Monday
  - 1 - 2pm Thursday
- TA study session (Manchester 121): 7-8pm Thursday

Warm-up question

- A roulette wheel has 38 slots numbered 00, 0, and 1–36. Two are green, 18 are red, and 18 are black.
- If a gambler bets based on color, the return on a $1 bet is $2
- A gambler has $50, and will continuously bet $1 on red until they double their money (have $100) or lose the money they came with
- What is the probability the gambler doubles their money?

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

Designing an experiment

Step 1: need a roulette wheel! (and money)

(38 slots, 2 g, 18 red, 18 black)

vector: $\underbrace{g, g}_{2}, \underbrace{r, r, \ldots, r}_{18}, \underbrace{b, b, \ldots, b}_{18}$

Step 2: spin the wheel!

Sample from our vector (roulette wheel)

Step 3: check result, and update our money

code? → { if spin is red : money = money + 1

if spin is not red : money = money − 1

Step 4: keep spinning until money = 100 or money = 0

(loop?)

Step 5: Repeat whole process, many times!

(estimate a probability)

(for loop)

Step 1: representing the roulette wheel

```r
wheel <- c(rep("green", 2), rep("black", 18), rep("red", 18))

wheel
```

vecter 2g 18 b 18 r

```
 [1] "green"  "green"  "black"  "black"  "black"  "black"  "black"  "black"
"black"
[10] "black"  "black"  "black"  "black"  "black"  "black"  "black"  "black"
"black"
[19] "black"  "black"  "red"    "red"    "red"    "red"    "red"    "red"
"red"
[28] "red"    "red"    "red"    "red"    "red"    "red"    "red"    "red"
"red"
[37] "red"    "red"
```

- `rep` repeats a value a specified number of times
- `c()` combines vectors into a single vector

Step 2: spin the wheel!

```r
spin <- sample(wheel, size = 1)

spin
```

```
[1] "red"
```

```
if    spin is    red:
          money =  money +1
if    spin is   not red:
          money  =  money -1
```

Step 3: change in money

```r
money <- 50
spin <- sample(wheel, size = 1)

if(spin == "red"){
  money <- money + 1
} else {
  money <- money - 1
}

spin
```

```
[1] "black"
```

```r
money
```

```
[1] 49
```

- if the result was red, gain a dollar
- otherwise, lose a dollar

Step 3: change in money

Another way of writing the conditional statement:

```r
money <- 50
spin <- sample(wheel, size = 1)

money <- ifelse(spin == "red", money + 1, money - 1)

spin
```

condition to check

if condition is true

otherwise do this

```
[1] "black"
```

```r
money
```

```
[1] 49
```

update money

Step 4: keep spinning

The gambler continues to bet until they have $0 or $100.

Question: Is a `for` loop appropriate for iterating the betting process?

for loop: repeats a chunk of code a fixed # of times

while loop: repeats code while a condition is true

Step 4: keep spinning

```r
money <- 50 # starting money

while(money > 0 & money < 100){
  spin <- sample(wheel, size = 1)
  money <- ifelse(spin == "red", money + 1, money - 1)
}

money
```

repeat until either money == 0
or money == 100

code to repeat while loop runs

```
[1] 0
```

- `while` loop: repeat the process until the condition is ~~true~~ false

(while condition is true)

Step 5: repeat the process

```
set.seed(279)
                                  # simulations
nsim <- 1000                  ←
results <- rep(NA, nsim)  ← vector to store results
                                  play the game nsim times
for(i in 1:nsim){         ←
  money <- 50 # starting money

  while(money > 0 & money < 100){
    spin <- sample(wheel, size = 1)
    money <- ifelse(spin == "red", money + 1, money - 1)
  }

  results[i] <- ...money == 100
}
```

- What should I check at each iteration?

results[i] : ith entry in results vector

$\frac{\text{sum(results)}}{\text{nsim}}$ or mean(results)

Step 5: repeat the process

```r
set.seed(279)

nsim <- 1000
results <- rep(NA, nsim)

for(i in 1:nsim){
  money <- 50 # starting money

  while(money > 0 & money < 100){
    spin <- sample(wheel, size = 1)
    money <- ifelse(spin == "red", money + 1, money - 1)
  }

  results[i] <- money == 100
}

mean(results)
```

```
[1] 0.008
```