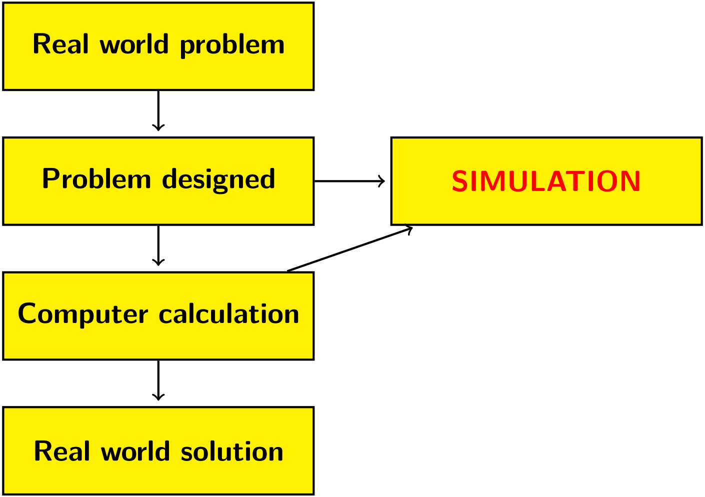
Part 3 Simulation with R

Dr. Nguyen Quang Huy

Dec 30, 2019

Why simulation.



Why simulation

Real world problem (uncertainty output)

↓

RANDOM SIMULATION

↓

Real world solution

Monte Carlo simulation

Monte Carlo (MC) simulation is based on the law of large number in probability theory:

Law of large number Let X_1, X_2, $\cdots$, is an infinite sequence of i.i.d random variables with expected value $E(X_1) = E(X_2) = \cdots = \mu$; the sample average

$$\bar{X}_n = \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$$

converges to the expected value

$$\bar{X}_n \to \mu \text{ for } n \to \infty$$

Why simulation

Example 1: Flip 100 fair coins simutaneously, how many coins show head (H) ? $\rightarrow$ design problem on computer:

Why simulation

Example 1: Flip 100 fair coins simutaneously, how many coins show head (H) ? $\rightarrow$ design problem on computer:

```
x<-sample(c(0,1),100,replace=TRUE) #goolge for sample functio
sum(x) # real world answer
```

```
## [1] 49
```

Why simulation

Example 1: Flip 100 fair coins simutaneously, how many coins show head (H) ? → design problem on computer:

```
x<-sample(c(0,1),100,replace=TRUE) #goolge for sample functio
sum(x) # real world answer
```

```
## [1] 49
```

Example 2: Flip 100 fair coins simutaneously, what the probability that the numbers of coin showing head (H) is larger than 65

Why simulation

Example 1: Flip 100 fair coins simutaneously, how many coins show head (H) ? → design problem on computer:

```
x<-sample(c(0,1),100,replace=TRUE) #goolge for sample function
sum(x) # real world answer
```

```
## [1] 49
```

Example 2: Flip 100 fair coins simutaneously, what the probability that the numbers of coin showing head (H) is larger than 65

```
y<-vector(mode="numeric",10^6)
for (i in 1:10^6){x<-sample(c(0,1),100,replace=TRUE)
y[i]<-sum(x)}
sum(y>65)/10^6
```

```
## [1] 0.00087
```

Why simulation

Example 3: Rolling 2 dies simutaneously, what is probability that the sum of 2 faces is less than or equal to 5 $\rightarrow$ EASY ...

Why simulation

Example 3: Rolling 2 dies simutaneously, what is probability that the sum of 2 faces is less than or equal to 5 $\rightarrow$ EASY ...

11	12	13	14	15	16
21	22	23	24	25	26
31	32	33	34	35	36
41	42	43	44	45	46
51	52	53	54	55	56
61	62	63	64	65	66

$\rightarrow$ answer is $\frac{10}{36} = \frac{5}{18}$

Why simulation

Example 3 (continue): Rolling 10 dies simutaneously, what is probability that the sum of 10 faces is less than or equal to 30 $\rightarrow$ NOT EASY $\rightarrow$ SIMULATION

Why simulation

Example 3 (continue): Rolling 10 dies simutaneously, what is probability that the sum of 10 faces is less than or equal to 30 $\rightarrow$ NOT EASY $\rightarrow$ SIMULATION

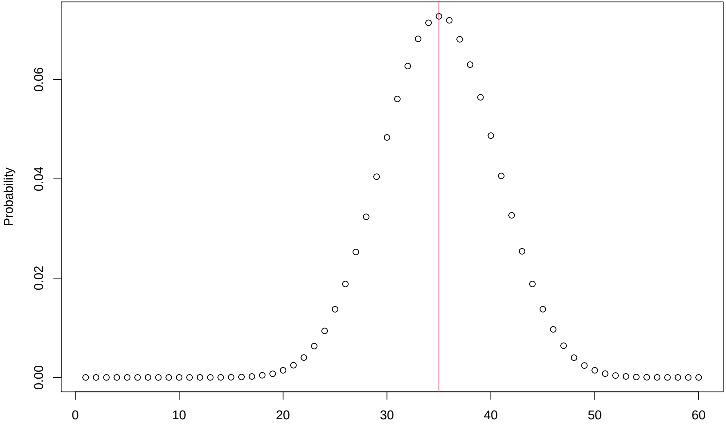
```
y<-vector(mode="numeric",10^6) # rolling 10 dies 1 million ti
for (i in 1:10^6){x<-sample(c(1,2,3,4,5,6),10,replace=TRUE)
y[i]<-sum(x)}
sum(y<=30)/10^6
```

```
## [1] 0.204038
```

Example 3 (continue): Rolling 10 dies simutaneously. Finding n such that

$$n = \arg\max_{6 \leq k \leq 60} \mathbb{P}\left(\text{sum of 10 faces} = k\right)$$

Why simulation



Why simulation

Example 4 In a card game where 52 cards are dealt evenly to 4 players. What is probability of Quads (Tu quy)

Why simulation

Example 4 In a card game where 52 cards are dealt evenly to 4 players. What is probability of Quads (Tu quy)

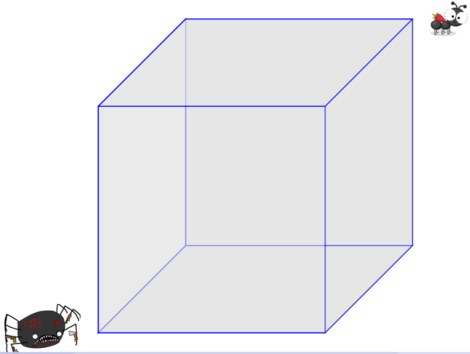
```
x<-c(1:13,1:13,1:13,1:13)
TestTuQuy<-function(y){
  h<-vector(mode="numeric",13)
  for (i in 1:13){h[i]<-sum(y==i)}
  ifelse(max(h)==4,1,0) }
Total<-0
for (i in 1:10){#chia bai 10 lan
j<-sample(1:52)
z<-x[j]
if (TestTuQuy(z[1:13])+TestTuQuy(z[14:26])+TestTuQuy(z[27:39])
Total
```

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

Simulation to solve probability problem

Problem 1: Blind spider

An ant and a blind spider are on opposite corners of a cube. The ant is stationary and the spider moves at random from one corner to another along the edges only. What is the expected number of turns before the spider reaches the ant ?



Simulation to solve probability problem

Problem 1: Blind spider

- Current position of the spider is (0,0,0) while position of the ant is (1,1,1).
- How does the spider randomly move ?
  - Current position of the spider is (x,y,z) with $x, y, z \in \{0, 1\}$
  - Next position of the spider is determined by randomly choosing between x, y, z and replace it by (1 - value of itself).
  - For example, the current position of the spider is (0,1,0). The sample function sample(1:3) returns value 1, the spider is moving as follows:

$$(\mathbf{0}, 1, 0) \to (\mathbf{1}, 1, 0)$$

- The spider reaches the ant if the current position of the spider is (1,1,1)

Simulation to solve probability problem

Problem 1: Blind spider How many steps that the spider needs to reach the ant:

```
SpiderPosition<-c(0,0,0)
AntPosition<-c(1,1,1)
MovingNumber<-0
while(sum(abs(SpiderPosition-AntPosition))>0){
    k<-sample(1:3,1)
    SpiderPosition[k]<-1-SpiderPosition[k]
    # print(SpiderPosition) # if you want to see the spider po
    MovingNumber<-MovingNumber+1
    }
MovingNumber
```

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

Simulation to solve probability problem

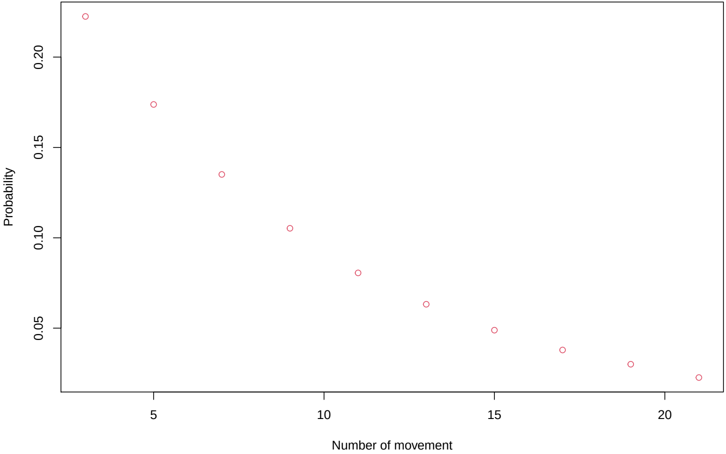
Problem 1: Blind spider Let the spider catches the ant 100,000 times (or more) to calculate the expected number of movement

```
Result<-vector(mode="numeric",10^5)
for (i in 1:10^5){
  SpiderPosition<-c(0,0,0)
  AntPosition<-c(1,1,1)
  MovingNumber<-0
  while(sum(SpiderPosition==AntPosition)<3){
    k<-sample(1:3,1)
    SpiderPosition[k]<-1-SpiderPosition[k]
    MovingNumber<-MovingNumber+1}
  Result[i]<-MovingNumber}
mean(Result)
```

```
## [1] 9.96394
```

Simulation to solve probability problem

Problem 1: Blind spider (continue) We can calculate the probability mass function of number of movement:



Simulation to solve probability problem

Problem 2 - Monkey typing: Suppose a monkey is typing randomly at a typewriter whose only keys are the capital letters A, D, M, U. What is the expected time it will take for the monkey to type the following words

- "MAD"
- "MUM"

Simulation to solve probability problem

```
KeyBoard<-c("A","D","M","U")
str<-KeyBoard[sample(1:4,3,replace=TRUE)]
MyString<-paste(str[1],str[2],str[3])
TypingNumber<-3
while (sum(str==c("M","A","D"))<3){ str[1]<-str[2]
    str[2]<-str[3]
    str[3]<-KeyBoard[sample(1:4,1)]
    MyString<-paste(MyString,str[3])
    TypingNumber<-TypingNumber+1}
TypingNumber
```

```
## [1] 222
```

```
substr(MyString,max(0,nchar(MyString)-20),nchar(MyString))
```

```
## [1] "D A M A U D M M M A D"
```

Simulation to solve probability problem

Problem 2 The monkey types until the word “MUM” appears:

```
result<-vector(mode="numeric",10)
for (i in 1:10){#10^4 when running
KeyBoard<-c("A","D","M","U")
str<-KeyBoard[sample(1:4,3,replace=TRUE)]
MyString<-paste(str[1],str[2],str[3])
TypingNumber<-3
while (sum(str==c("M","U","M"))<3){ str[1]<-str[2]
    str[2]<-str[3]
    str[3]<-KeyBoard[sample(1:4,1)]
    MyString<-paste(MyString,str[3])
    TypingNumber<-TypingNumber+1}
result[i]<-TypingNumber}
mean(result) # "MUM": 68 and "MAD": 64
```

```
## [1] 92.5
```

Simulation to solve probability problem

Problem 3 - Drunk passenger(s): You are at the end of a line of 100 airline passengers is waiting to board a plane. The n^{th} passenger in line has ticket for the seat number n. Being drunk, the first person in line picks a random seat (equally likely for each seat). All of the other passengers will go to their proper seats unless it is already occupied; If it is occupied, they will then find a free seat to sit in, at random. What is the probability that you will sit in your proper seat ? What does this probability change if there are k drunk passengers ?

Simulation to solve probability problem

Problem 3 - Drunk passenger(s)

- Step 1: First k passengers take k seats randomly.
- Step 2: Let passengers $(k+1)$, $(k+2)$, $\cdots$, 99 board respectively, if his proper seat is occupied he will choose an empty seat randomly.
- Step 3: If passenger 100 seats on 100th seat, return 1 and return 0 otherwise.
- Step 4: Repeat the procedure N times ($N = 10^5$ for example) and calculated the expectation of returns.

Simulation to solve probability problem

Problem 3 - Drunk passenger(s)

```
Seat<-vector(mode="numeric",100)
SeatAvailable<-1:100
Seat[1]<-sample(1:100,1) #Seat of Drunk man
SeatAvailable<-SeatAvailable[-Seat[1]] #Remove seat 1
for (i in 2:100){
    if (sum(SeatAvailable==i)==0){
      Seat[i]<-sample(SeatAvailable,1)}
    else{Seat[i]<-i}
    SeatAvailable<-SeatAvailable[-Seat[i]]# Remove seat i
}
Seat[100]
```

```
## [1] 13
```

Simulation to solve probability problem - Drunk passenger

```
result<-vector(mode="numeric",10)#using 10^5 simulation in yo
for (j in 1:10){
  Seat<-vector(mode="numeric",100)
  SeatAvailable<-1:100
  Seat[1]<-sample(1:100,1) #Seat of Drunk man
  SeatAvailable<-SeatAvailable[-Seat[1]] #Remove seat 1
  for (i in 2:100){
      if (sum(SeatAvailable==i)==0){
        Seat[i]<-sample(SeatAvailable,1)}
      else{Seat[i]<-i}
      SeatAvailable<-SeatAvailable[-Seat[i]]}
  result[j]<-Seat[100]}
  result
```

```
##  [1] 100  10 100  26 100 100  42  41  33 100
```

Simulation to solve probability problem

Problem 3 - Drunk passenger(s) What is the probability that you will sit on your proper seat if 1^{st} passenger, 2^{nd} passenger, $\cdots$, k^{th} passenger are drunk ?

Simulation to solve probability problem

Problem 4 - Messing with envelops: There are n letters and n envelopes. Your servant puts the letters randomly in the envelopes so that each letter is in one envelope and all envelopes have exactly one letter. Calculate the expected number of envelopes with correct letter inside them.

Simulation to solve probability problem

Problem 4 - Messing with envelops: There are *n* letters and n envelopes. Your servant puts the letters randomly in the envelopes so that each letter is in one envelope and all envelopes have exactly one letter. Calculate the expected number of envelopes with correct letter inside them.

```
n<-20
x<-1:n
for (i in 1:10){#Using 10^6 simulations
  y<-sample(1:n,n,replace=FALSE)
  result[i]<-sum(x==y)}
result
```

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

```
mean(result)
```

```
## [1] 0.4
```

Simulation to solve probability problem

Problem 5 - Tennis tournement A tennis tournament has $k = 4$ levels and $n = 2^k$ players. At the begining, random pairs are formed and one player from each pair proceeds to next level. Given that player number i has better skill than player j if $i < j$ and the one with better skills always wins.

- What is the probability that a player outside the top 5 plays the final match.
- Your skill is 5, calculate the probability that you will play in Semi-final.

Simulation to solve "tennis tournement" problem

```r
k<-4
position<-1:(2^k)
player<-sample(1:(2^k),2^k,replace=FALSE)
for (i in 1:(k-1)){# (k-1) levels, not final match yet
  OutTour<-c(0)
  for (j in 1:2^(k-i)){
    a<-ifelse(player[2*j-1]>player[2*j],2*j-1,2*j)
    OutTour<-c(OutTour,a)}
  player<-player[-OutTour[2:(2^(k-i)+1)]]
}
player # Final match
```

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

Simulation to solve "tennis tournement" problem

```
k<-4
position<-1:(2^k)
result<-vector(mode="numeric",10)
for (n in 1:10){
player<-sample(1:(2^k),2^k,replace=FALSE)
for (i in 1:(k-1)){# (k-1) levels, not final match yet
  OutTour<-c()
  for (j in 1:2^(k-i)){
    a<-ifelse(player[2*j-1]>player[2*j],2*j-1,2*j)
    OutTour<-c(OutTour,a)}
  player<-player[-OutTour]
}
if (max(player)>5){result[n]<-1}}
mean(result)
```

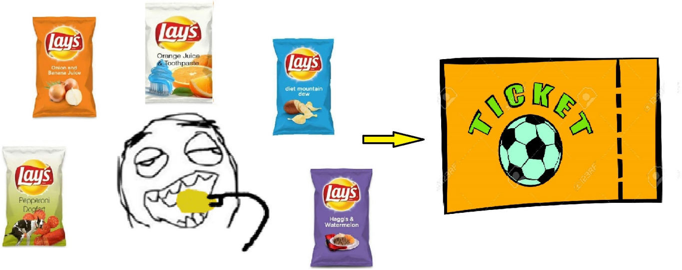
```
## [1] 0
```

Simulation to solve “tennis tournement” problem

Calculate the probability that you, with skill is equal to 5, will play in Semi-final ?

Simulation to solve probability problem

Problem 6 - Collecting lucky coupons: Pepsico company is holding a contest where everyone who collects one each of 10 different coupons wins a football ticket. You get a coupon with each purchase of a Lay's chips packet, and each coupon is equally likely. Whats the expected number of packets you have to eat in order to get a ticket ?



Simulation to solve probability problem

```
k<-5
YourCoupon<-vector(mode="numeric",5)
Eating<-c()
EatNumber<-0
while(sum(YourCoupon)<5){
    NewOne<-sample(1:k,1)
    Eating<-c(Eating,NewOne)
    YourCoupon[NewOne]<-1
}
length(Eating)
```

```
## [1] 21
```

```
Eating
```

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

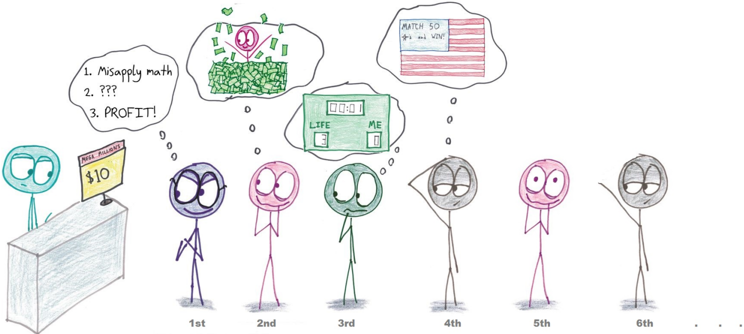
Simulation to solve probability problem

```
k<-5
result<-vector(mode="numeric",10)
for (i in 1:10){#using 10^6 simulation
YourCoupon<-vector(mode="numeric",5)
Eating<-c()
EatNumber<-0
while(sum(YourCoupon)<5){
    NewOne<-sample(1:k,1)
    Eating<-c(Eating,NewOne)
    YourCoupon[NewOne]<-1
}
result[i]<-length(Eating)}
mean(result)
```

```
## [1] 13.1
```

Simulation to solve probability problem

Problem 7 - Winning a free movies ticket: At a movie theater, the manager announces that they will give a free ticket to the first person in line whose birthday is the same as someone who has already bought a ticket. You have the option of getting in line at any time. Assuming that you don't know anyone else's birthday. What position in line gives you the greatest chance to win the free ticket?



Simulation to solve probability problem

```
result<-vector(mode="numeric",10)# recommend 10^6 simulations
for(i in 1:10){
birthday<-c()
newbirthday<-sample(1:365,1)
position<-1
while(sum(birthday==newbirthday)==0){
    birthday<-c(birthday,newbirthday)
    newbirthday<-sample(1:365,1)
    position<-position+1}
result[i]<-position}
result[1:5]
```

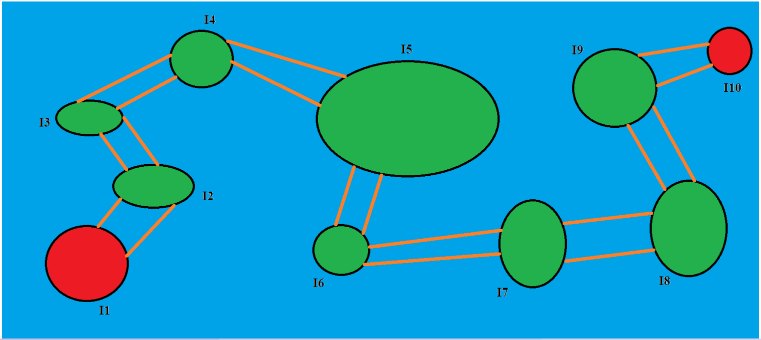
```
## [1] 39 14 18 17 11
```

Simulation to solve probability problem

Problem 8 - All die or one survive: In a room stand $n = 100$ armed and angry people. At each chime of a clock, everyone simultaneously spins around and shoots a random other person. The persons shot fall dead and the survivors spin and shoot again at the next chime. Eventually, either everyone is dead or there is a single survivor. What is the probability that there will be a survivor.

Simulation to solve probability problem

Problem 9 - Inlands and bridge: There are 10 inlands I_1, I_2, $\cdots$, I_{10}, and you want to travel from inland I_1 to inland I_{10}. For each j = 1, 2, . . . ,9, inlands I_j and I_{j+1} are connected by 2 bridges. These bridges are looked like to each other but if you use the bad one, it will break down immediately and you have to swim back to inland I_1. What is expected number of bridges you have to pass to go to inland I_{10} ?



Simulation to solve probability problem

Problem 10 - Points on circle: What is the probability that n random points on a circle are lying on a half of the circle.