MTH511a - 2020: Lecture 1

Instructor: Dootika Vats

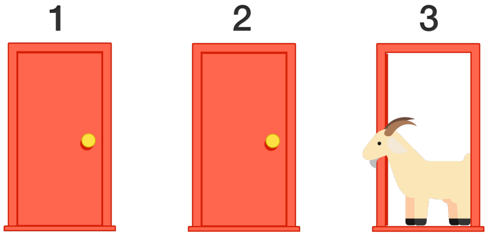
The instructor of this course owns the copyright of all the course materials. This lecture material was distributed only to the students attending the course MTH511a: "Statistical Simulation and Data Analysis" of IIT Kanpur, and should not be distributed in print or through electronic media without the consent of the instructor. Students can make their own copies of the course materials for their use.

1 Introduction to Monte Carlo

We will learn many things about Monte Carlo in this course. However, this short introduction is meant to familiarize you with the concept of using simulation to "answer" questions.

Sometimes, answers to certain mathemtical questions are not obtainable by standard/known calculations. Monte Carlo methods, or Monte Carlo experiments, are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. The idea is use the computer to "simulate" or "mimic" the premises of the question and then based on what the computer returns, guess the answer. This of course, sounds too vague and general, so let's work through a few examples.

Example 1: Monty Hall Problem (aka Khul Ja Sim Sim)



> You are on a game show, being asked to choose between three doors. One door has a car, and the other two have goats. After you choose a door, the host, Monty Hall, opens one of the other doors, which he knows has a goat behind it. Monty then asks whether you would like to switch your choice of door to the other remaining door. Do you choose to switch or not to switch?

Of course, whether you will switch or not depends on which action has the largest probability of winning the car (unless you like goats more than cars!). Now at first glance it seems like it would not matter whether you swtiched or not. However, this is not the case! We can answer this question mathematically, but you may not believe the answer.

Instead, let's try and simulate this situation on a computer. We will write an R code to repeat a Monty Hall experiment multiple times. And in each time, we will see whether swtiching or not swtiching would be more beneficial.

```r
set.seed(1)
repeats <- 1e4 # We will repeat the experiment 10000 times
win.no.switch <- numeric(length = repeats) # will save 0 or 1 based on winning no switch
win.switch <- numeric(length = repeats)   # will save 0 or 1 based on winning switch

for(r in 1:repeats) # Repeat process many times
{
  # The setup
  doors <- 1:3    # three doors
  prize <- sample(1:3, 1)  # randomly select the door which has the prize

  # Contestants are ready. Game starts
  chosen.door <- sample(1:3, 1)   # choose a door

  # reveal a door that is not the chosen door and not the door with a prize in it
  # doing rep(., 2) because sample() is being annoying
  which.reveal <- rep((1:3)[-c(prize, chosen.door)], 2)
  reveal <- sample(which.reveal, size = 1) # randomly choose which door to reveal

  win.no.switch[r] <- chosen.door == prize  #tracking win if don't change door

  chosen.door <- (1:3)[-c(reveal, chosen.door)] #change door
  win.switch[r] <- chosen.door == prize # tracking win if change door
}
```

In the above code `win.no.switch` and `win.swtich` contains 10000 1s or 0s depending on whether the player would have won by not switching or switching respectively. To see which option is better, we can look at the mean of those two vectors:

```r
mean(win.no.switch) #Prob of winning if you don't switch
```

```
## [1] 0.331
```

```r
mean(win.switch) # Prob of winning if you switch
```

```
## [1] 0.669
```

Voila! The estimated probability of winning if you stay with your door is .3375, but if you switch, it is .6625! It looks like switching is more beneficial! And indeed, you can show mathematically that

$$\Pr\left(\text{Winning if you switch the door}\right) = \frac{2}{3}\,.$$

You'd think a Monte Carlo solution is not needed for this simple problem. But evidently, it was needed in the early 90s to verify the solution. There is a fascinating story to the Monty Hall problem, how it came about, and how it confused mathematicians all over the world [Vos Savant, 1997].

Example 2: Toy Collector Problem

Children (and some adults) are frequently enticed to chips packets in an effort to collect toys found in these packets. Assume there are 15 different kinds of toys and each packet contains exactly one with each toy having probability

Figure	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Probability	.2	.1	.1	.1	.1	.1	.05	.05	.05	.05	.02	.02	.02	.02	.02

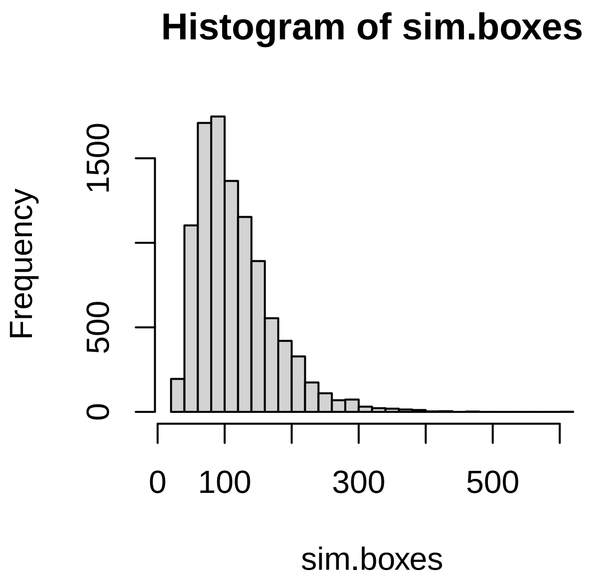
Q. What is the expected number of packets needed to collect all 15 action figures.

Of course, this is a question of great practical importance since it allows us to know what number of chips packets we will need to buy. Surprisingly, the solution to this problem is mathematically quite complicated. A Monte Carlo solution is often the best bet.

```
set.seed(1)

# the setup
prob.table <- c(.2, .1, .1, .1, .1, .1, .05, .05, .05, .05, .02, .02, .02, .02, .02)
boxes <- 1:length(prob.table)

# writing the code a little differently now.
# I made a function that will be called repeatedly
box.count <- function(prob)
{
  check <- rep(0, length(prob))
  i <- 0
  while(sum(check) < length(prob)) # check if all toys collected
  {
    x <- sample(boxes, 1, prob = prob) # generate a toy with given prob
    check[x] <- 1    # x has been collected
    i <- i + 1
  }
  return(i)
}

repeats <- 1e4
sim.boxes <- numeric(repeats)
for(i in 1:repeats)
{
  sim.boxes[i] <- box.count(prob = prob.table)
}
```

Let X = number of chips packets needed to collect all toys. Then X is a random variable, and in the above code `sim.boxes` are 10000 IID realizations of this random variable. We can then, for example, look at the empirical distribution of this random variable:

```
hist(sim.boxes, breaks = 30)
```



From the looks of it X has a skewed distribution with about a 100 chips packets being sufficient most of the times, and in rare circumstances we may need 400-500 chips packets. We can estimate the expected number of chips packages by taking the average:

```r
mean(sim.boxes)
```

```
## [1] 116.4749
```

The estimated average number of chips we need to buy in order to obtain at least one of each of the toy is 116.47.

Example 3: Complicated Integral

Consider the integral

$$\theta = \int_0^{\pi} e^{\sin(x)}\, dx\,.$$

The above integral does not have a standard analytical form (although a solution exists in terms of the Bessel function). But suppose we are interested in calculating θ. In Monte Carlo simulation, we take this "simple" mathematical problem with no variability and turn it into a statistical problem. That is, instead of "calculating" θ exactly, we will "estimate" θ.

Let $Y \sim U(0, \pi)$. Then Y has the following probability density function:

$$f(y) = \frac{1}{\pi} I(0 < y < \pi)\,.$$

Notice that we can multiply and divide by $f(y)$ in θ, so that

$$\theta = \int_0^{\pi} e^{\sin(x)}\, dx = \pi \int_0^{\pi} \frac{1}{\pi} e^{\sin(x)}\, dx = \pi \int_0^{\pi} e^{\sin(x)} f(x) dx = \pi \mathrm{E}\left[e^{\sin(x)}\right]\,,$$

where the last expectation is with respect to $U(0, \pi)$. Thus, the integral problem is now explicitly an expectation problem. In order to estimate θ, we can generate sample from $U(0, \pi)$ repeatedly, calculate $e^{\sin(x)}$ for each draw and then take an average.

```r
set.seed(1)
repeats <- 1e4
```

```
esin <- numeric(length = repeats)
for(i in 1:repeats)
{
  samp <- runif(1, min = 0, max = pi) # draw from U(0, pi)
  esin[i] <- exp(sin(samp))
}
pi * mean(esin)  #pi*E(exp(sin(x)))
```

```
## [1] 6.18
```

Thus we can conclude that $\theta \approx 6.18$. We couldn't calculate θ exactly, but we could estimate it!

2 The Statistics of Monte Carlo

In each of the three examples, there were three steps

1. identify a distribution that requires sampling
   - **Monty Hall** - Bernoulli(p) where p was unknown
   - **Toy Collector** - Distribution of the number of chips packets required
   - **Integral** - Uniform$(0, \pi)$
2. draw random samples from some this distribution
   - **Monty Hall** - using the Monty Hall game setup
   - **Toy Collector** - by simulating the buying of chips packets
   - **Integral** - `runif` function
3. calculate mean of some particular function of interest
   - **Monty Hall** - identity function
   - **Toy Collector** - identity function
   - **Integral** - $e^{\sin(x)}$

Keeping this in mind, the statistics of Monte Carlo are then simple. Let F be a target distribution identified in Step 1 such that it is defined on some support $\mathcal{X}$ and has a density f. Let $g : \mathcal{X} \to \mathbb{R}$ be a function such that

$$\theta = \int_{\mathcal{X}} g(x)f(x)dx \,,$$

is of interest. The above assumes a continuous random variable. For discrete random variables, the distribution function F is asssumed to have a probability mass function $p(x)$ such that

$$\theta = \sum_{x_i \in \mathcal{X}} g(x_i)p(x_i)$$

is of interest.

In order to estimate θ, suppose that we have a mechanism to draw $X_1, \dots, X_n \overset{iid}{\sim} F$. Then an estimate of θ is

$$\hat{\theta} := \frac{1}{n}\sum_{t=1}^{n} g(X_t) \,.$$

Q. Why is $\hat{\theta}$ a good estimator?

The simple answer is the weak law of large numbers, which says that as long as $\theta < \infty$

$$\hat{\theta} := \frac{1}{n}\sum_{t=1}^{n} g(X_t) \xrightarrow{p} \theta \qquad \text{as } n \to \infty\,,$$

where $\xrightarrow{p}$ denotes convergence in probability.

Recall: Weak law of large numbers. Let $X_1, \dots, X_n$ be a sequence of iid random variables having mean $\mu < \infty$. Then for any $\epsilon > 0$,

$$\Pr\left\{\left|\frac{X_1 + \cdots + X_n}{n} - \mu\right|\right\} \to 0 \text{ as } n \to \infty.$$

Q. How do we generate iid samples from complex or even simple distributions?

This will be the question that we will focus on for the first few weeks of the course. However, keep in mind these examples to motivate *why* we need to generate samples from distributions.

3 Questions to think about

1. What if θ is not finite? What happens then?
2. How should we choose n? In my simulations, I set ‘repeats = 1e4‘. Is that a good choice? Sould I do more? Could I have done less?
3. How can we do Monte Carlo in the third example if the bounds on the integral were 0 to 2π?

References

[Vos Savant, 1997] Vos Savant, M. (1997). *The power of logical thinking: Easy lessons in the art of reasoning... and hard facts about its absence in our lives.* Macmillan.