Chapter 3

Simulation and Monte Carlo integration

In this chapter we introduce the concept of generating observations from a specified distribution or sample, which is often called Monte Carlo generation.

The name of *Monte Carlo* was applied to a class of mathematical methods first by scientists working on the development of nuclear weapons in Los Alamos in the 1940s. For history of Monte Carlo see Kalos and Whitelock (1986), *Monte Carlo Methods*, Vol. I: Basics, Wiley.

3.1 The law of large numbers

There are two types of laws of large numbers, a strong one and a weak one. They may be thought of as the theoretical background for Monte Carlo methods. Below is a strong law of large numbers.

Theorem 1. If $X_1, X_2, \ldots, X_n, \ldots$ are independent and identically distributed random variables with $EX_k = \mu, k = 1, 2, \ldots,$ then

$$P\left(\lim_{n\to\infty} \frac{\sum_{k=1}^n X_k}{n} = \mu\right) = 1.$$

Proof. See page 85 of Billingsley, P. (1995), *Probability and Measure*, 3rd edition, Wiley. □

The next one is a weak law of large numbers.

Theorem 2. If $X_1, X_2, \ldots, X_n, \ldots$ are independent and identically distributed random variables with $EX_k = \mu$ and $E(X_k - \mu)^2 = \sigma^2, k = 1, 2, \ldots,$ then, for any positive constant ε,

$$\lim_{n\to\infty} P\left(\left|\frac{\sum_{k=1}^n X_k}{n} - \mu\right| \geq \varepsilon\right) = 0.$$

Proof. It is not so hard, and is left as a homework.

3.2 The "hit-or-miss" method

Consider an event A whose probability is p. Define a sequence of independent and identically distributed Bernoulli random variables $X_1, X_2, \ldots, X_n$ by

$$X_k = \begin{cases} 1, & \text{if event } A \text{ occurs,} \\ & \\ 0, & \text{otherwise,} \end{cases} \qquad k = 1, 2, \ldots$$

You may imagine $\{X = 1\}$ as "hit" and $\{X = 0\}$ as "miss".

The moment estimator and maximum likelihood estimator of p based on $X_1, X_2, \ldots, X_n$ are the same, with

$$\hat{p} = \frac{\sum_{k=1}^{n} X_k}{n}.$$

This estimator is unbiased, that is, $E\hat{p} = p$. More importantly, it follows from the strong law of large numbers that

$$P\left(\lim_{n\to\infty} \frac{\sum_{k=1}^{n} X_k}{n} = p\right) = 1.$$

The earliest documented use of random sampling to find the solution to an integral seems to be that of Comte de Buffon, which is often called Buffon's needle problem. Here we estimate π by the "hit-or miss" method.

Example 1 (Estimation of π). Suppose that a disk with radius 1 meter is put inside a square whose length is 2 meters. We toss a little needle to hit the square. The question is: what is the probability that the needle lies within the disk?

The answer is $\frac{\pi}{4}$.

To perform the Monte Carlo method, let U_1 and U_2 be independent random variables uniformly distributed on the interval $[-1, 1]$, and they may be treated as two sides of a square. Define a random variable that represents the event of hitting the disk,

$$X = \begin{cases} 1, & \text{if } U_1^2 + U_2^2 \leq 1, \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$EX = P(U_1^2 + U_2^2 \leq 1) = \frac{\pi}{4}.$$

For a sequence of such i.i.d. variables, $X_1, \ldots, X_n$, by the strong law of large numbers we obtain

$$\lim_{n\to\infty} \frac{\sum_{k=1}^{n} X_k}{n} = \frac{\pi}{4},$$

with probability 1.

```
pimc=function(n){                                 #  n is the number of simulations
                u1=runif(n, -1, 1)                #  generate a uniform variable U_1
                u2=runif(n, -1, 1)                #  U_2
                x=rep(0, n)                       #  format of X
                x[u1^2 + u2^2 <=1]=1              #  X variable
                pimc=4*mean(x)                    #  pi estimator
                pimc
                }
> pimc(10)
  [1] 4
> pimc(100)
  [1] 3.28
> pimc(1000)
  [1] 3.06
> pimc(10000)
  [1] 3.136
> pimc(100000)
  [1] 3.13792
> pimc(1000000)
  [1] 3.141824
```

3.3 Monte Carlo integration

Suppose that $g(x)$, $x \in [0,1]$, is a real and continuous function. The question is: how to estimate the integral $\int_0^1 g(x)dx$? There may be many approximations for this integral. Here we use the Monte Carlo method.

Example 2 (Estimation of π). Consider the function

$$g(u) = 4\sqrt{1-u^2}, \qquad u \in [0,1].$$

It is easy to show that $\int_0^1 g(u)du = \pi$.

To use the Monte Carlo method, suppose that $U_1, U_2, \ldots, U_n$ are independent random variables uniformly distributed on the interval $[0,1]$. Then $X_1 = g(U_1), X_2 = g(U_2), \ldots, X_n = g(U_n)$ are i.i.d random variables, with mean

$$EX_k = Eg(U_k) = \int_0^1 g(u)du = \pi.$$

```
pimc=function(n){                      #  n is the number of simulations
            u=runif(n)                 #  generate a sequence of uniform U random numbers
            g=4*sqrt(1-u^2)            #  g(U)
            pimc=mean(g)               #  pi estimator
            pimc
            }
```

3.4 Generating random numbers

This section describes some approaches to generate random variables or vectors with specified distributions. When talking about "generate" a random object, we mean an algorithm whose output is an object of the desired type.

To generate a random variable (or vector) having a target distribution $F(x)$ on $\mathbb{R}^d$, we typically start from a sequence of i.i.d. uniform random numbers. Thus, our task is: given an i.i.d. sequence $U_1, U_2, \ldots$ that follow an uniform distribution on the interval $[0,1]$, find $m \geq d$ and a deterministic function $g : [0,1]^m \to \mathbb{R}^d$ such that the distribution function of $g(U_1, \ldots, U_m)$ is $F(x)$.

3.4.1 The inverse method

Theorem. Let X be a random variable with distribution function $F(x)$.

(1) For every $t \in [0,1]$,

$$P(F(X) \leq t) \leq t.$$

In particular, if $F(x)$ is continuous on the real line, then $U = F(X)$ is uniformly distributed on $[0,1]$.

(2) Define the inverse of F as

$$F^{-1}(y) = \inf\{x, F(x) \geq y\}, \qquad 0 < y < 1.$$

If U is a uniform random variable on $[0,1]$, then $F^{-1}(U)$ has distribution function $F(x)$.

Example 1. An exponential density function is of the form

$$f(x) = \begin{cases} \lambda \exp(-\lambda x), & x > 0, \\ 0, & x \leq 0, \end{cases}$$

where λ is a positive constant. The cumulative distribution function

$$F(x) = \begin{cases} 1 - \exp(-\lambda x), & x > 0, \\ 0, & x \leq 0, \end{cases}$$

is strictly increasing and continuous on $[0, \infty)$, with the inverse

$$F^{-1}(x) = -\frac{1}{\lambda} \ln(1 - x), \qquad 0 < x < 1.$$

Thus, we could use the following procedure to generate exponential random numbers.

```
rexp =  function(n, lambda){
         u = runif(n)                  # generate uniform random numbers
         g =  -1/lambda*log(1-u)       # exponential random numbers
         g
         }
```

One way to see whether a data set comes from a particular distribution is the q-q plot, which is based on sample order statistics (or sample quantiles) and theoretical quantiles. For a set of observations $x_1, x_2, \ldots, x_n$, we put them in order,

$$x_{1:n} \le x_{2:n} \le x_{n:n}.$$

If the target distribution function is $F(x)$, then the quantile for $\frac{k}{n+1}$ is $F^{-1}\left(\frac{k}{n+1}\right)$, $k = 1, \ldots, n$. A q-q plot is a scatterplot of points

$$\left(x_{k:n}, \quad F^{-1}\left(\frac{k}{n+1}\right)\right), \quad k = 1, \ldots, n.$$

These points would be close to a straight line if the data set actually comes from $F(\cdot)$.

Below is an example for an exponential case.

```
x = rexp(50, 1)                         # generate 50 exponential random numbers
sq = sort(x)                            # sample quantiles
tq = function(u){-log(1-u)}             # exponential quantile function with lambda=1
plot(sq, tq(1:50/51),
     xlab="Sample quantiles", ylab="Theoretical quantiles")
```

The next example is for generating normal observations, but not using the inverse method.

Example 2 (Generating normal observations). To simulate normal random variables, Box and Muller (1958) suggested the following procedure. Let U_1 and U_2 be independent random variables uniformly distributed on the interval $(0, 1)$. Define X_1 and X_2 by

$$\begin{aligned} X_1 &= \sqrt{-2\ln U_1}\cos(2\pi U_2), \\ X_2 &= \sqrt{-2\ln U_1}\sin(2\pi U_2). \end{aligned}$$

It can be shown that X_1 and X_2 are independent standard normal random variables.

```
rnormal=function(n){
    u1=runif(n)
    u2=runif(n)
    x = (-2*log(u1))^(1/2)*cos(2*pi*u2)
    x
    }

qqnorm(rnormal(100))
qqline(rnormal(100))
```

3.4.2 Rejection method

We want to generate random numbers from a target distribution with the density $f(x)$.

Suppose that $f(x)$ can be rewritten as

$$f(x) = \frac{h(x)g(x)}{\int_{-\infty}^{\infty} h(u)h(u)du}, \quad x \in \mathbb{R},$$

where $h(x)$ is a nonnegative function, and $g(x)$ is another density function that has a simple form. The rejection procedure is:

Step 1. Generate $Y \sim g(x)$.

Step 2. Generate $U \sim uniform(0,1)$.

Step 3. Accept Y is $U \leq h(Y) \leq 1$, otherwise go back to Step 1.

There are two discrete events involved in the above procedure: a success (Y is accepted), and no success (Y is rejected). To confirm the procedure makes sense, we need evaluate the conditional probability

$$P(Y \leq x \mid success).$$

Note that

$$\begin{aligned}
P(Y \leq x \text{and success}) &= P(Y \leq x, U \leq h(Y) \leq 1) \\
&= \int\int_{y \leq x, u \leq h(y) \leq 1} (\text{joint density of } Y \text{ and } U)\ dydu \\
&= \int\int_{y \leq x, u \leq h(y) \leq 1} g(y) \cdot 1 dydu \\
&= \int_{y \leq x} \left(\int_{u \leq h(y) \leq 1} du \right) g(y)dy \\
&= \int_{y \leq x} h(y)g(y)dy \\
&= \int_{-\infty}^{x} h(y)g(y)dy, \qquad x \in \mathbb{R},
\end{aligned}$$

from which we obtain

$$P(success) = \lim_{x \to \infty} P(Y \leq x \text{ and success}) = \int_{-\infty}^{\infty} h(y)g(y)dy.$$

Thus,

$$\begin{aligned}
P(Y \leq x \mid success) &= \frac{P(Y \leq x \text{ and success})}{P(success)} \\
&= \frac{\int_{-\infty}^{x} h(y)g(y)dy}{\int_{-\infty}^{\infty} h(y)g(y)dy}, \qquad x \in \mathbb{R}.
\end{aligned}$$

This means the above procedure works theoretically.

Example 1. Let

$$f(x) = \begin{cases} \frac{4}{\pi}\frac{1}{1+x^2}, & 0 < x < 1, \\ 0, & \text{otherwise.} \end{cases}$$

We will use the rejection method to generate random numbers from this distribution.

Choose $g(x)$ to be a uniform density,

$$g(x) = \begin{cases} 1, & 0 < x < 1, \\ 0, & \text{otherwise,} \end{cases}$$

and $h(x) = f(x)$. Then, the algorithm is

```
re =function(n){
  h=function(x){4/pi*(1+x^2)^(-1)}
  y = runif(n)
  u = runif(n)
  re = y[u <= h(y) & h(y) <=1]
  #if(u>h(y)) repeat
  re
  }
```

```
> re(100)
  [1] 0.7310012 0.9566798 0.8679352 0.5366552 0.5633315 0.5500107 0.8593855
  [8] 0.6485496 0.6402452 0.6262020 0.7063684 0.5277953 0.7371513 0.7603023
 [15] 0.5445079 0.6313800 0.6478479 0.8180368 0.8224039 0.6279190 0.7134841
 [22] 0.7110942 0.8868061 0.6479279 0.8643747 0.5624662 0.6680021 0.9070305
 [29] 0.8095055 0.6669217 0.5366229 0.6524184 0.9968572 0.8017481 0.5706411
 [36] 0.9855979 0.7278967 0.9353859 0.8450351 0.7762552 0.9016270 0.9033395
 [43] 0.5256584
```

Example 2. Consider the following algorithm that was proposed by Marsaglia and Bray[1]:

(1) Generate U and V independent uniform $(-1, 1)$ random variables.

(2) Set $W = U^2 + V^2$.

(3) If $W > 1$, go to Step (1).

(4) Set $Z = \sqrt{(-2\ln W)/W}$, and let $X_1 = UZ$ and $X_2 = VZ$.

Show that the random variables X_1 and X_2 are independent and normally distributed with mean zero and variance 1. Use this algorithm to generate 100 normal random numbers, and obtain a normal Q-Q plot.

[1] Marsaglia and Bray (1964), A convenient method for generating normal variables, SIAM Review, vol. 6,260-264.

3.5 Importance sampling

Suppose that we have an n-dimensional integral

$$I = \int_{\mathbb{R}^n} g(\mathbf{x}) f(\mathbf{x}) d\mathbf{x}$$

that needs to evaluate, where $f(\mathbf{x})$ is a density function. The Monte Carlo procedure is to take a random sample $\mathbf{X}_1, \ldots, \mathbf{X}_n$ from this distribution, and then to form the mean

$$g_n = \frac{1}{n} \sum_{k=1}^{n} g(\mathbf{X}_k).$$

When $\int_{\mathbb{R}^n} g^2(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} < \infty$, the strong large number law asserts that

$$\frac{1}{n} \sum_{k=1}^{n} g(\mathbf{X}_k) \to I,$$

with probability 1.

Since g_n estimates I, we can write

$$g_n = I + error.$$

This error is a random variable, with mean 0 and variance

$$var(g_n) = \frac{1}{n} \left\{ \int_{\mathbb{R}^n} g^2(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} - I^2 \right\}.$$

What we want is to reduce the error (variance).

But, $f(\mathbf{x})$ is not necessarily the best density function to use in the Monte Carlo simulation even though it appears in the integrand. A different density function, $\tilde{f}(\mathbf{x})$, can be introduced into the integral as follows:

$$I = \int_{\mathbb{R}^n} \frac{g(\mathbf{x}) f(\mathbf{x})}{\tilde{f}(\mathbf{x})} \tilde{f}(\mathbf{x}) d\mathbf{x},$$

where $\tilde{f}(\mathbf{x}) \geq 0$, $\int_{\mathbb{R}^n} \tilde{f}(\mathbf{x}) d\mathbf{x} = 1$, and $\frac{g(\mathbf{x}) f(\mathbf{x})}{\tilde{f}(\mathbf{x})} < \infty$ except for perhaps a (countable) set of points. How to choose an appropriate $\tilde{f}$ is a trick in importance sampling.