Introduction to simulation and Monte Carlo methods

Eric Gaussier

University Grenoble Alps
UFR-IM^{2}AG
Eric.Gaussier@imag.fr

Table of content

Illustrative example

Many practical applications requires the evaluation of the probability $P(a < X \leq b)$ (waiting time, survival time, ...)

- Let F denote the cumulative distribution (CDF) function of X: $F(x) = P(X \leq x)$
- Then: $P(a < X \leq b) = F(b) - F(a)$
- However, the function F is not always known in closed-form, even though the density function f is (e.g., the normal distribution)
  The density function f of X is defined, when it exists, as: $F(x) = \int_{-\infty}^{x} f(t)dt$
- When f exists: $P(a < X \leq b) = \int_{a}^{b} f(t)dt$

Illustrative example (cont'd)

- Let us assume that $X \sim N(0, 1)$. In other words:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} exp(-\frac{(x-\mu)^2}{2\sigma^2})\ (\mu = 0, \sigma = 1)$$

- F is not known and we cannot compute exactly the above integrals
- However, let us assume that we know how to generate points from f. What can we do?

Illustrative example (cont'd)

Example code in *R*

```
*******
 set.seed12               # seed for random generation
 m = 50000                # num. of observations sampled
 a = ?; b = ?
 z = rnorm(m)             # m random obs. from std normal
 mean(z > a & z ≤ b)      # prop. of obs. in (a, b]
*******
```

Structure of the course

- Generating random numbers
- Ordinary Monte Carlo and limit theorems
- Markov chains
- MCMC: Monte Carlo Markov Chains

Table of content

Linear congruential generator

- A linear congruential generator produces integers r_i iteratively acc. to:

  $$r_{i+1} = ar_i + b(mod\ d)$$

  for integers $a > 0$, $b \leq 0$ and $d > 0$. a is called the **multiplier**, b the **increment** and d the **modulus** of the generator.
- The generation process is started with a positive **seed**, $s = r_1 < d$, often taken from a computer clock
- If $b = 0$, the generator is called **multiplicative**
- Linear congruential generators can shuffle integers from $0, 1, 2, \cdots, d-1$ in ways that *look* random

Linear congruential generator (cont'd)

```
*******
 d=53; a=20; b=0; s=21                 # modulus, mult., incr.
 m = 60                                # length of run
 r[1] = s                              # set seed
 for (i in 1:(m-1))
       r[i+1] = (a * r[i] + b) %% d    # gen. random int.
*******
```

$\rightarrow$ 21 49 26 43 12 28 30 17 ...

Linear congruential generator (cont'd)

- The above generator runs through all 52 numbers before it repeats; its **period** is 52; it has **full period**
- With $a = 23 \rightarrow 21\ 36\ 2\ 47\ 21 \cdots$
- Fit into the interval $(0, 1)$: $u_i = (r_i + 0.5)/d$
- Properties of good generators:
  1. Large modulus and full (or at least large) period
  2. Uniform distribution (histogram of values consistent with UNIF(0,1))
  3. Independent structure (pairwise independence – plot of the pairs (u_i, u_{i+1}) should fill the unit square

Transformations of uniform random variables

Let us assume that we have a "good" generator for UNIF(0,1). UNIF(0,1) is such that:

$$P(0 < X \leq a) = a,\ \ a \leq 1;\ \ F(x) = x,\ \ x \in [0,1]$$

Theorem: *Suppose X is a continuous random variable with CDF F_X. Then the random varaibles $Y = 1 - F_X(X)$ and $Z = F_X(X)$ are distributed as UNIF(0,1), also noted U[0,1]*

Transformations of uniform random variables (cont'd)

Proof:

$$\begin{aligned} F_Z(z) &= P(Z \leq z) = P(F_X(X) \leq z) \\ &= P(X \leq F_X^{-1}(z)) = F_X(F_X^{-1}(z)) \\ &= z \end{aligned}$$

Remark: F^{-1} is called the **quantile** transformation

Transformations of uniform random variables (cont'd)

Quantile transformation When the quantile function F^{-1} has a formula, to simulate a value of $X \sim F$:

1. Simulate a value of $U \sim U[0, 1]$
2. Use $X = F^{-1}(U)$

Transformations of uniform random variables (cont'd)

Accept-reject Suppose X has density f, from which one can not directly simulate, and F^{-1} has no known formula. Let us further assume that we can construct another density $g(x)$ from which it is easy to simulate and such that $\frac{f(x)}{g(x)}$ is uniformly bounded. Then, we can simulate X as follows:

1. Find a density function g and constant c: $\frac{f(x)}{g(x)} \leq c, \ \forall x$
2. Generate $U \sim U[0, 1]$
3. Generate $X \sim G$
4. Retain X if $U \leq \frac{f(X)}{cg(X)}$
5. Repeat till required number of samples

Transformations of uniform random variables (cont'd)

Accept-reject Why does it work?

Theorem: *Let $X \sim g$ and $U \sim U[0, 1]$. Then, the conditional density of X given that $U \leq \frac{f(X)}{cg(X)}$ is f*

Simulating from common distributions

- **Bernoulli** To generate $X \sim Ber(p)$, generate $U \sim U[0,1]$ and set $X = I_{U>1-p}$
- **Binomial** To generate $X \sim Bin(n,p)$, generate $X_1, ..., X_n \sim Ber(p)$ and set $X = \sum_{i=1}^{n} X_i$
- **Standard exponential** To generate $X \sim Exp(1)$ $(exp(-x))$, generate $U \sim U[0,1]$ and set $X = -\log(U)$
- **Standard normal** To generate $X \sim N(0,1)$, use the accept-reject method with $g(x) = \frac{1}{2} exp(-|x|)$ and $c = \sqrt{\frac{2e}{\pi}}$

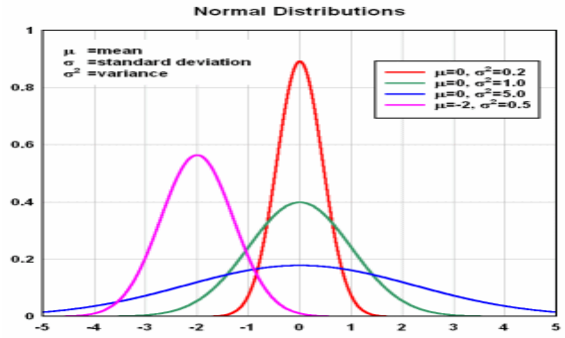
Summary (part 1)

1. Generation of random integers with linear congruential generators
2. From random integers to $U[0,1]$
3. From $U[0,1]$ to common distributions via quantile transformation and accept-reject

Table of content

Ordinary Monte Carlo

Illustration $X \sim N(0,1)$; $P(0 < X \leq 1)$?



Ordinary Monte Carlo (cont'd)

First solution (integration)

```
*******
 set.seed12                  # seed for random generation
 m = 50000                   # num. of observations sampled
 a = 0; b = 1
 w = (b - a)/m
 u = a + (b - a) * runif(m)  # vector of m random points
 h = dnorm(u)                # hts of density above u
 sum(w*h)                    # approx. prob.
*******
```

$\rightarrow$ *0.3413*

Ordinary Monte Carlo (cont'd)

Second solution (acceptance-rejection method)

```
*******
 set.seed12                          # seed for random generation
 m = 50000                           # num. of observations sampled
 u = runif(m, 0, 1)
 h = runif(m, 0, 0.4)                height of "upper" rectangle
 frac.acc = mean(h < dnorm(u)
 0.4 * frac.acc
*******
```

$\rightarrow$ *0.3410*

Ordinary Monte Carlo (cont'd)

Why does the above approach work?
SLLN: Strong Law of Large Numbers

Theorem: *Suppose X_1, X_2, ..., X_n are i.i.d random variables with a finite mean μ. Then:*

$$P(\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} X_i = \mu) = 1 \;\; (\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i)$$

(and $\frac{1}{n}\sum_{i=1}^{n} \phi(X_i)$ converges almost surely to $E[\phi(X)]$)

Ordinary Monte Carlo (cont'd)

The central limit theorem is used to compute confidence intervals for the approximate values obtained with Monte Carlo.

Theorem: *Suppose X_1, X_2, ..., X_n are i.i.d random variables with finite mean μ and finite variance σ^2. Then, for large n:*

$$\sqrt{n}(\bar{X}_n - \mu) \sim N(0, 1)$$

Summary (part 2)

1. Ordinary Monte Carlo can be used to compute many (complex) integrals, provided the function is known
    - Expectation and prob. on intervals are special cases (integrals)
    - Different methods (incl. sampling from the underlying distribution for prob. on intervals)
2. SLLN guarantees the convergence while CLT provides confidence intervals
3. What to do when the density function is known up to a constant ($f(x) = \frac{h(x)}{c}$)?

Table of content

Definitions and notations

Definition 1 A sequence of random variables $X_0, ..., X_n$ is said to be a *(finite state) Markov chain* for some state space S if for any $x_{n+1}, x_n, ..., x_0 \in S$:

$$P(X_{n+1} = x_{n+1} | X_0 = x_0, ..., X_n = x_n) = P(X_{n+1} = x_{n+1} | X_n = x_n)$$

X_0 is called the initial state and its distribution the initial distribution

Definition 2 A Markov chain is called homogeneous or stationary if $P(X_{n+1} = y | X_n = x)$ is independent of n for any x, y

Definition 3 Let $\{X_n\}$ be a stationary Markov chain. The probabilities $p_{ij} = P(X_{n+1} = j | X_n = i)$ are called the *one-step transition probabilities*. The associated matrix P is called the *transition probability matrix*

Definitions and notations (cont'd)

Definition 4 Let $\{X_n\}$ be a stationary Markov chain. The probabilities $p_{ij}^{(n)} = P(X_{n+m} = j | X_m = i)$ are called the *n-step transition probabilities*. The associated matrix $P^{(n)}$ is called the *transition probability matrix*

Remark: P is a stochastic matrix

Theorem (Chapman-Kolgomorov equation) Let $\{X_n\}$ be a stationary Markov chain and $n, m \geq 1$. Then:

$$p_{ij}^{m+n} = P(X_{m+n} = j | X_0 = i) = \sum_{k \in S} p_{ik}^m p_{kj}^n$$

Example

Two urns: urn 1 contain balls A and B; urn 2 contains balls C and D. In each successive trial , randomly chose a ball from urn 1 and urn 2 and switch them

Regularity (ergodicity)

Definition 5 Let $\{X_n\}$ be a stationary Markov chain with transition probability matrix P. It is called *regular* if there exists $n_0 > 0$ such that $p_{ij}^{(n_0)} > 0 \ \forall i, j \ \in S$

Theorem (fundamental theorem for finite Markov chains) Let $\{X_n\}$ be a regular, stationary Markov chain on a state space S of t elements. Then, there exists π_j, $j = 1, 2, ..., t$ such that:

(a) For any initial state i, $P(X_n = j | X_0 = i) \rightarrow \pi_j$, $j = 1, 2, ..., t$

(b) The row vector $\pi = (\pi_1, \pi_2, ..., \pi_t)$ is the unique solution of the equations $\pi P = \pi, \ \pi 1' = 1$

(c) Any row of P^r converges towards π when $r \rightarrow \infty$

Remark: π is called the long-run or stationary distribution

Summary (part 3)

1. Stationary, regular Markov chains admit a stationary (steady-stable) distribution
2. This distribution can be obtained in different ways (as in the PageRank computation)
3. Power method: let the chain run for a sufficiently long time!

Table of content

MCMC overview

Problem Posterior probability estimation in which denominator is intractable and prior distribution known up to a constant:

$$\underbrace{P(\theta|x)}_{\text{posterior}} = \frac{\overbrace{f(x|\theta)}^{\text{likelihood}} \; \overbrace{\pi(\theta)}^{\text{prior}}}{\underbrace{\int_{\Theta} f(x|\theta)\pi(\theta)d\theta}_{m(x)}}$$

$m(x)$ is a normalizing constant, often unknown; it is also an expectation!

MCMC overview (ont'd)

MCMC steps:

1. Identify the target distribution π and the state space S (support of π) from which one wants to simulate
2. Construct Markov chain $\{X_n\}$ with stationary distribution π
3. Run the chain for a sufficiently long time and use the values $X_{B+1}, ..., X_n$ as samples for B large
4. Approximate $E_\pi[\phi(X)]$ by $\frac{1}{(n-B)} \sum_{k=B+1}^{n} \phi(X_k)$

Step 2?

Reversible Markov chains

Definition 5 Let $\{X_n\}$ be a stationary Markov chain with transition probability matrix P and state space S. The chain is said to be *reversible* if there exists a nonnegative function $\pi(x)$ on S such that: $p_{ij}\pi(i) = p_{ji}\pi(j)$

Remark: if π is a steady-state, then
$P(X_n = j|X_{n+1} = i) = P(X_{n+1} = j|X_n = i)$

Theorem Let $\{X_n\}$ be a regular, reversible, stationary Markov chain on a state space S. Then π (reversibility) is a unique stationary distribution of the chain and
$\frac{1}{n}\sum_{k=1}^{n}\phi(X_k) \to E_\pi[\phi(X)]$

Metropolis-Hastings algorithm

Metropolis

$$\begin{aligned} p_{ij} &= \alpha_{ij}\gamma_{ij},\ i,j \in S,\ j \neq i \\ p_{ii} &= 1 - \sum_{j\neq i} p_{ij} \end{aligned} \tag{1}$$

Metropolis-Hastings (special case)

$$\alpha_{ij} = c;\ \gamma_{ij} = \min\{1, \frac{\pi(j)}{\pi(i)}\}$$

Is the reversibility condition satisfied?

Metropolis-Hastings algorithm (cont'd)

Theorem The general Metropolis algorithm has π as its stationary distribution

Remark: $0, 1, ..., B$ is the *burn-in period* before the chain stabilizes (B is either set sufficiently large or is obtained through stabilization criteria)

Example

$Y \sim Geo(p)$ ($P(Y = x) = p(1-p)^{x-1}$) and one wants to simulate from the conditional distribution of Y given that $Y \leq n$, where $n > 1$ is a specified integer

Then:

$$\pi(x) = \frac{p(1-p)^{x-1}}{1-(1-p)^n} I_{1 \leq x \leq n}$$

p_{ij}?

The Gibbs sampler

When state space is large (of dimension m), previous methods may be difficult to apply. Gibbs sampling iterates the process:

From state $x = (x_1, ..., x_m)$ go to state $y = (y_1, ..., y_m)$ via $(y_1, x_2, ..., x_m)$, $(y_1, y_2, x_3, ..., x_m)$, ...

The transition: $(x_1, ..., x_m) \rightarrow (y_1, x_2, ..., x_m)$ is made by simulating from the distribution $\pi(x_1|x_2, ..., x_m)$. Such conditional distributions are called *full conditionals*

The Gibbs sampler (cont'd)

The Gibbs sampler makes its transition by changing one coordinate at a time

- Random scan Gibbs sampler
- Systematic Gibbs sampler

Full conditional: $\pi(x_i|x_{-i})$; transition from x to $y = (x_1, ..., x_{i-1}, y_i, x_{i+1}, ..., x_m)$ acc. to $\pi(x_i|x_i)$

The Gibbs sampler (cont'd)

In the random scan, the transition probability matrix is: $P = \frac{1}{m}(P_1 + ... + P_m)$ where P_i has entries given by:

$$p_{i,xy} = \pi(y_i|x_{-i}) I_{y_j = x_j \, \forall \, j \neq i}$$

Each P_i is reversible ($\pi(x)p_{i,xy} = \pi(y)p_{i,yx}$), and so is P; same holds for regularity

Reversibility may not be ensured for systematic scan; it is however ensured for the *order randomized systematic scan Gibbs sampler* (choose order at random then systematic scan)

The Gibbs sampler (cont'd)

Theorem Suppose $\pi(x) > 0$ for all $x \in S$. Then the random scan and the order randomized systematic scan Gibbs samplers both have π has unique stationary distribution

Remarks

1. The systematic Gibbs sampler often used in practice (speed issues)
2. Burn-in period to consider prior to averaging

Example Gibbs sampling from Dirichlet distributions

Summary (part 4)

1. Reversible Markov chains to simulate in situations where previous methods fail
2. Metropolis algorithm (Metropolis-Hastings version)
3. Gibbs sampling (with random and systematic scans)

Table of content

Conclusion

- Generic methods that can be applied in many different cases
- Sometimes methods are combined (Gibbs sampling with Metropolis-Hastings, e.g. for a particular full conditional)
- Algorithms relatively easy; the definition of the chain slightly more difficult for MCMC
- Widespread use, but not necessarily the fastest methods
- Confidence intervals (reliability measures)

Some references used to prepare the course

- E. Suess, B. Trumbo. Introduction to Probability Simulation and Gibbs Sampling with R, Springer, 2010
- A. DasGupta. Probability for Statistics and Machine Learning, Springer, 2011