Monte Carlo Methods

Juan F. Rubio-Ramírez

Emory University

Why Monte Carlo?

From previous chapter, we want to compute:

1. Posterior distribution:

$$\pi(\theta \mid Y^T, i) = \frac{f(Y^T \mid \theta, i)\, \pi(\theta \mid i)}{\int_{\Theta_i} f(Y^T \mid \theta, i)\, \pi(\theta \mid i)\, d\theta}$$

2. Marginal likelihood:

$$P(Y^T \mid i) = \int_{\Theta_i} f(Y^T \mid \theta, i)\, \pi(\theta \mid i)\, d\theta$$

Difficult to assess analytically or even to approximate (Phillips, 1996). Resort to simulation.

A Bit of Historical Background and Intuition

Metropolis and Ulam (1949) and von Neumann (1951). Why the name “Monte Carlo”?
Two simple examples:

1. Probability of getting a total of six points when rolling two fair dice.
2. Throwing darts at a graph.

Classical Monte Carlo Integration

Assume we know how to generate draws from $\pi(\theta \mid Y^T, i)$. What does it mean to draw from it? Two basic questions:

1. Why do we want to do it?
2. How do we do it?

Why Do We Do It? Strong Reason

Basic intuition: Glivenko–Cantelli's theorem. Let $X_1, \ldots, X_n$ be i.i.d. with cdf F and empirical cdf $F_n(\cdot, \omega)$. Then, as $n \to \infty$,

$$\sup_{-\infty < x < \infty} \left| F_n(x, \omega) - F(x) \right| \xrightarrow{\text{a.s.}} 0.$$

Can be generalized to include dependence (van der Vaart & Wellner, 1997).

Why Do We Do It? Weak Reason

Let $h(\theta)$ be a function of interest. By the Law of Large Numbers:

$$\mathbb{E}_{\pi(\cdot|Y^T,i)}[h(\theta)] \approx h_m := \frac{1}{m}\sum_{j=1}^{m} h(\theta_j).$$

If $\mathrm{Var}_\pi[h(\theta)] < \infty$, then by the CLT:

$$\mathrm{Var}_\pi[h_m] \approx \frac{1}{m}\sum_{j=1}^{m}\big(h(\theta_j) - h_m\big)^2.$$

How Do We Do It? Random Number Generators

Large literature. Two good surveys:

- Devroye (1986) *Non-Uniform Random Variate Generation*.
- Robert & Casella (2004) *Monte Carlo Statistical Methods*.

Random Draws?

Natural sources of randomness are hard to use. A computer is deterministic! von Neumann (1951): "Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."

Basic Building Block

- Pseudo-random number generators are highly non-linear iterative algorithms that "look random".
- We focus on $U(0, 1)$ draws.
- In general, other distributions arise from transforming uniforms.

Goal

Design iterative algorithms (Lehmer, 1951) that:

1. Are unpredictable for the uninitiated (relation to chaotic dynamical systems).
2. Pass standard statistical tests (K–S, ARMA(p, q), etc.).

Basic Component: Congruential Generators

Multiplicative congruential generator:

$$x_i = (ax_{i-1} + b) \mod (M + 1) \text{ and } x_0 \text{ is the seed.}$$

Then:

$$u_i = x_i/(M + 1) \text{ is an } U(0, 1)$$

Example: Generating Integers and Uniforms

Parameters: $a = 5$, $b = 3$, $M = 16$, seed $x_0 = 7$.

$$x_i = (ax_{i-1} + b) \bmod (M+1) \quad \Rightarrow \quad x_i \in \{0, \ldots, M\}.$$

i	Computation	x_i	$u_i = x_i/(M+1)$
0	seed	7	0.412
1	$(57 + 3) \bmod 17 = 4$	4	0.235
2	$(54 + 3) \bmod 17 = 6$	6	0.353
3	$(56 + 3) \bmod 17 = 16$	16	0.941
4	$(516 + 3) \bmod 17 = 15$	15	0.882

Choices of Parameters

Period/performance hinge on a, b, M. Bad choice example: $a = 13$, $c = 0$, $M = 31$, $x_0 = 1$ (historical bad examples: IBM RND, 1960s).

A Good Choice

Traditional: $a = 7^5 = 16807$, $c = 0$, $m = 2^{31} - 1$. Period bounded by M. 32 vs 64 bit hardware matters. Beware IEEE floating-point standard. Alternatives exist.

Real Life

Don't code your own RNG. MATLAB implements state-of-the-art (e.g., KISS by Marsaglia & Zaman, 1991). For Fortran/C++: see DIEHARD battery.

Nonuniform Distributions

We often need non-uniform draws. Basic approach, move from uniforms via:

- Transformations (standard tricks).
- Inverse cdf method.

These underpin commercial software.

Transformations Example: Normal via Box–Muller

Let $U_1, U_2 \sim U(0,1)$. Then

$$x = \cos(2\pi U_1)\sqrt{-2\log U_2}, \quad y = \sin(2\pi U_1)\sqrt{-2\log U_2}$$

are i.i.d. $N(0,1)$ (points lie on a spiral in (x,y)).

Transformations Example: Multivariate Normal

If $x \sim N(0, I)$ and $\Sigma\Sigma^\top$ is the covariance, then

$$y = \mu + \Sigma x \sim N(\mu, \Sigma\Sigma^\top).$$

Use Cholesky for Σ.

The Inverse Transform Method

Goal: Generate a random variable X with a known CDF using uniform draws.

- Let:
$$X = F^{-1}(U) \text{ and } U \sim U(0,1)$$
- Then
$$P(X \leq x) = P(F^{-1}(U) \leq x) = P(U \leq F(x)) = F(x).$$
- Hence X has distribution F.

Example Inverse 1: Exponential Distribution

Let $X \sim \text{Exp}(\lambda)$ with

$$F(x) = 1 - e^{-\lambda x}, \qquad x \geq 0.$$

Solve for x in terms of $u = F(x)$:

$$u = 1 - e^{-\lambda x} \quad \Rightarrow \quad x = -\frac{1}{\lambda} \ln(1 - u).$$

Algorithm

1. Draw $U \sim U(0,1)$.
2. Set $X = -\frac{1}{\lambda} \ln U$.

Then X follows $\text{Exp}(\lambda)$. Simple and exact.

Example Inverse 2: Discrete Case (Bernoulli)

Let $X \in \{0, 1\}$ with $P(X = 1) = p$. CDF:

$$F(x) = \begin{cases} 0, & x < 0, \\ 1 - p, & 0 \leq x < 1, \\ 1, & x \geq 1. \end{cases}$$

Algorithm:

$$X = \begin{cases} 1, & \text{if } U < p, \\ 0, & \text{otherwise.} \end{cases}$$

Comment: works for any discrete distribution by comparing U with cumulative probabilities.

Fundamental Theorem of Simulation

- Transformations and Inverse Method very limited set of distributions.
- We now present a general approach
- Suppose $f(x)$ is a probability density on a measurable space $\mathcal{X} \subseteq \mathbb{R}^d$.
- Imagine the set under its graph:

$$A = \{(x, y) \in \mathcal{X} \times [0, \infty) : 0 \leq y \leq f(x)\}.$$

- If we draw points uniformly in A, the projection of these points onto the x-axis follows exactly the distribution with density $f(x)$.
- Intuitively, higher parts of the curve receive proportionally more projected points.

Fundamental Theorem of Simulation

Theorem. Let $f : \mathbb{R}^d \to [0, \infty)$ satisfy $\int f(x)\,dx = 1$. Define

$$A = \{(x, y) \in \mathbb{R}^d \times [0, \infty) : \, 0 \leq y \leq f(x)\}.$$

If (X, Y) is uniformly distributed on A, marginal distribution of X has density $f(x)$.

Idea: Uniform sampling under the density surface produces samples distributed according to that density.

Proof

- Since $\int f(x)\,dx = 1$, the total volume of A is one:

$$|A| = \int_{\mathbb{R}^d} \int_0^{f(x)} dy\,dx = 1.$$

- The joint density of (X, Y) is therefore

$$p_{X,Y}(x, y) = \mathbf{1}\{0 \leq y \leq f(x)\}.$$

- Integrating out y,

$$p_X(x) = \int_0^{f(x)} 1\,dy = f(x).$$

- Hence the marginal of X has density $f(x)$. $\square$

Acceptance Sampling: Motivation

We cannot easily draw points uniformly under $f(x)$.

- Introduce a simpler density $g(x)$ such that we can draw from it.
- Find a constant $a > 0$ such that:

$$f(x) \leq a\, g(x) \quad \forall x.$$

- The function $a\, g(x)$ is called an **envelope** of $f(x)$.

Idea: sample uniformly under $a\, g(x)$ and keep only points under $f(x)$.

Acceptance Sampling: Algorithm

1. Draw $X \sim g(x)$.
2. Draw $U \sim U(0,1)$.
3. Accept X if $U \leq \dfrac{f(X)}{a\,g(X)}$.

Interpretation:

- The pair $(X, U a g(X))$ is uniformly distributed under the curve $ag(x)$.
- Accepted draws correspond to points that lie under $f(x)$.
- Hence accepted X follow the target density $f(x)$.

Choice of a

Let $f, g \geq 0$ and assume $\operatorname{supp}(f) \subseteq \operatorname{supp}(g)$ (i.e., $g(x) = 0 \Rightarrow f(x) = 0$). Define

$$a = \sup_x \frac{f(x)}{g(x)}.$$

Claim: $f(x) \leq a\, g(x)$ for all x.

Is $a \geq 1$ Always?

Let f, g be densities with $\text{supp}(f) \subseteq \text{supp}(g)$ and

$$a = \sup_x \frac{f(x)}{g(x)}.$$

Claim. $a \geq 1$, with $a = 1$ iff $f = g$ almost everywhere (w.r.t. g). Hence $a > 1$ whenever $f \neq g$ on a set of positive g-measure.

Proof

Let $R(x) = \frac{f(x)}{g(x)}$ where $g(x) > 0$ (define $R = 0$ when $g = 0$; then $f = 0$ there). Since f, g are densities,

$$\mathbb{E}_g[R(X)] \; = \; \int \frac{f(x)}{g(x)} \, g(x) \, dx \; = \; \int f(x) \, dx \; = \; 1.$$

If $R(x) < 1$ for all x with $g(x) > 0$, then $\mathbb{E}_g[R(X)] < 1$, a contradiction. Thus $\sup_x R(x) \geq 1$, i.e. $a \geq 1$.

Moreover, $a = 1$ iff $R(x) \leq 1$ a.e. and $\mathbb{E}_g[R] = 1$, which forces $R(x) = 1$ a.e., i.e. $f(x) = g(x)$ a.e. $\square$

Quality of the Envelope

- The acceptance rate is $P(\text{accept}) = \dfrac{1}{a}$.
- A tight envelope (a close to 1) $\Rightarrow$ more efficient sampling.
- A loose envelope (a large) $\Rightarrow$ many rejections.
- Ideally, $g(x)$ resembles $f(x)$ in shape and tails.

Proof $P(\text{accept}) = \frac{1}{a}$

Let $A = \{U \leq f(X)/(ag(X))\}$. Then

$$\mathbb{P}(A) = \mathbb{E}[\mathbf{1}_A] = \mathbb{E}\left[\mathbf{1}\left\{U \leq \frac{f(X)}{ag(X)}\right\}\right] = \mathbb{E}\left[\mathbb{E}\left(\mathbf{1}\left\{U \leq \frac{f(X)}{ag(X)}\right\} \,\middle|\, X\right)\right].$$

Independence $\Rightarrow$ given $X = x$, $U \sim U(0,1)$ so

$$\mathbb{P}\left(U \leq \frac{f(X)}{ag(X)} \,\middle|\, X\right) = \frac{f(X)}{ag(X)}.$$

Hence

$$\mathbb{P}(A) = \mathbb{E}\left[\frac{f(X)}{ag(X)}\right] = \int \frac{f(x)}{ag(x)}\, g(x)\, dx = \frac{1}{a} \int f(x)\, dx = \frac{1}{a}.$$

Proof (One Line with the Supremum Property)

Define $r(x) = \frac{f(x)}{g(x)}$ for $g(x) > 0$ and set $r(x) = 0$ when $g(x) = 0$ (using $g(x) = 0 \Rightarrow f(x) = 0$). By definition of the supremum,

$$r(x) \ \leq \ \sup_z r(z) \ = \ a \quad \text{for every } x.$$

Multiplying by $g(x) \geq 0$ yields

$$f(x) \ = \ r(x)\, g(x) \ \leq \ a\, g(x) \qquad \forall x.$$

$\square$

Truncated Densities and Accept-reject

- Suppose the target is a **truncated version** of an easy distribution $g(x)$:

$$f(x) = \frac{g(x)\,\mathbf{1}\{x \in A\}}{P_g(A)}, \qquad A = \text{allowed region.}$$

- We can draw $X \sim g$ easily, but we only want $X \in A$.
- Accept–Reject Sampling fits perfectly:
  1. Draw $X \sim g$.
  2. If $X \in A$, accept; otherwise, reject.

Why It Works So Well

- Because $a = \frac{1}{P_g(A)}$, the acceptance probability is simply

$$P(\text{accept}) = P_g(X \in A).$$

- Efficiency depends only on how much mass of g lies inside A.
- If truncation is mild (e.g. 80–90% of the mass kept), then the acceptance rate is high and the algorithm is almost costless.
- Even for more severe truncation, the method is simple, exact, and needs no renormalization.

Key idea: For truncated densities, Accept–Reject $\Rightarrow$ "Draw from the full g and keep what's valid." No extra math—just logical filtering.

Truncated Distributions and the Accept–Reject Rule

Target density: truncated version of an easy $g(x)$,

$$f(x) = \frac{g(x)\,\mathbf{1}\{x \in A\}}{P_g(A)}, \qquad P_g(A) = \int_A g(x)\,dx.$$

Hence

$$\frac{f(x)}{g(x)} = \begin{cases} \dfrac{1}{P_g(A)}, & x \in A, \\ 0, & x \notin A. \end{cases}$$

To satisfy $f(x) \leq a\,g(x)$ for all x, choose

$$a = \sup_x \frac{f(x)}{g(x)} = \frac{1}{P_g(A)}.$$

Substitute into the Acceptance Condition

The generic rule:

$$U \leq \frac{f(X)}{a\, g(X)}.$$

Substitute the truncated expressions:

$$\frac{f(X)}{a\, g(X)} = \begin{cases} \dfrac{1/P_g(A)}{1/P_g(A)} = 1, & X \in A, \\ 0, & X \notin A. \end{cases}$$

Therefore

$$U \leq \begin{cases} 1, & X \in A, \\ 0, & X \notin A. \end{cases}$$

Interpretation:

- If $X \in A$: $U \leq 1$ always true $\Rightarrow$ accept.

Conclusion: Why It Simplifies Perfectly

- For truncated densities, the acceptance test reduces to a simple membership check:

$$U \leq \frac{f(X)}{a\,g(X)} \iff X \in A.$$

- Inside A, f and g have the same shape—just rescaled by $1/P_g(A)$.
- The acceptance rate is $P_g(A)$: the mass of g inside A.
- Hence the algorithm is extremely efficient:
  1. Draw $X \sim g$,
  2. Accept if $X \in A$.

Summary: In the truncated case, Accept–Reject becomes "keep draws that lie in the truncation region."

Acceptance Pitfalls

1. Many rejections: minimize a.
2. Need π/g bounded $\Rightarrow g$ must have thicker tails.
3. Computing a can be hard.

Can we do better? Yes—importance sampling.

Importance Sampling I

Same setup. For any integrable h,

$$\mathbb{E}_\pi[h(\theta)] = \int h(\theta)\frac{\pi(\theta)}{g(\theta)}g(\theta)\,d\theta.$$

Importance Sampling II

With draws $\{\theta_j\}_{j=1}^{m}$ from g,

$$h_m^{IS} := \frac{1}{m}\sum_{j=1}^{m} h(\theta_j)\,\frac{\pi(\theta_j)}{g(\theta_j)} \;\to\; \mathbb{E}_\pi[h(\theta)].$$

If $\mathbb{E}_\pi\left[\frac{\pi(\theta)}{g(\theta)}\right]$ exists, then

$$m^{1/2}\left(h_m^{IS} - \mathbb{E}_\pi[h(\theta)]\right) \Rightarrow N(0, \sigma^2),$$

with

$$\sigma^2 \approx \frac{1}{m}\sum_{j=1}^{m}\left(h(\theta_j) - h_m^{IS}\right)^2 \left(\frac{\pi(\theta_j)}{g(\theta_j)}\right)^2.$$

Importance Sampling IV: Variance Intuition

We want the weight ratio $\pi(\theta)/g(\theta)$ to be as flat as possible. Ideally $g = \pi$.

Importance Sampling V: Picking g

Use a local (e.g., Taylor) approximation to π as g. Question: how to compute the Taylor approximation?

Existence Condition

A simple sufficient condition: $\pi(\theta)/g(\theta)$ bounded. Denote $\omega(\theta) = \pi(\theta)/g(\theta)$.

Unknown Normalizing Constants

If only unnormalized densities $\tilde{\pi}, \tilde{g}$ are available, then

$$\mathbb{E}_{\pi}[h(\theta)] = \frac{\displaystyle\int h(\theta)\,\frac{\tilde{\pi}(\theta)}{\tilde{g}(\theta)}\,\tilde{g}(\theta)\,d\theta}{\displaystyle\int \frac{\tilde{\pi}(\theta)}{\tilde{g}(\theta)}\,\tilde{g}(\theta)\,d\theta}.$$

Self-Normalized IS Estimator

$$h_m^{SN} = \frac{\sum_{j=1}^{m} h(\theta_j)\,\omega(\theta_j)}{\sum_{j=1}^{m} \omega(\theta_j)}, \quad \sigma^2 \approx \frac{m \sum_{j=1}^{m} \left(h(\theta_j) - h_m^{SN}\right)^2 \omega(\theta_j)^2}{\left(\sum_{j=1}^{m} \omega(\theta_j)\right)^2}.$$

Example I: Bad g (Heavy Tails Target)

Suppose π is t_ν but we sample from $g = N(0,1)$. Estimate $\mathbb{E}[X]$ of t_ν using IS weights $\omega(\theta) = t_\nu(\theta)/\phi(\theta)$.

Computation Sketch

Draw $\theta_j \sim N(0,1)$, then

$$\widehat{\text{mean}} = \frac{1}{m}\sum_{j=1}^{m} \theta_j \, \omega(\theta_j), \quad \widehat{\text{Var}}(\widehat{\text{mean}}) = \frac{1}{m}\sum_{j=1}^{m} (\theta_j - \widehat{\text{mean}})^2 \, \omega(\theta_j)^2.$$

Illustration (Reported in Slides)

As $\nu = 3, 4, 10, 100$, the estimated variance of the mean falls, but can be extremely large for small ν under normal g.

Table (Example I)

ν	3	4	10	100
Est. Mean	0.1026	0.0738	0.0198	0.0000
Est. Var(Est. Mean)	684.52	365.66	36.82	3.59

Example II: Heavy-Tailed g for Light-Tailed Target

Now $\pi = N(0,1)$, but draw from $g = t_\nu$. Variance of the IS estimator is moderate across ν.

Table (Example II)

t_ν	3	4	10	100
Est. Mean	-0.0104	-0.0075	0.0035	-0.0029
Est. Var(Est. Mean)	2.0404	2.1200	2.2477	2.7444

Relative Numerical Efficiency (RNE)

If $g = \pi$, then

$$\sigma^2 \approx \frac{1}{m}\sum_{j=1}^{m}\left(h(\theta_j) - h_m^{IS}\right)^2 \approx \mathrm{Var}_\pi[h(\theta)].$$

Define

$$\mathrm{RNE} = \frac{\mathrm{Var}_\pi[h(\theta)]}{\sigma^2}.$$

RNE close to 1: good IS; near 0: poor IS.

RNE Tables

Target t_ν, $g = N(0,1)$:

$$\text{RNE} = \{0.0134,\ 0.0200,\ 0.0788,\ 0.2910\} \text{ for } \nu = \{3, 4, 10, 100\}.$$

Target $N(0,1)$, $g = t_\nu$:

$$\text{RNE} = \{0.4777,\ 0.4697,\ 0.4304,\ 0.3471\}.$$

Importance Sampling & Prior Robustness

Two researchers share the likelihood $f(Y^T \mid \theta)$ but have priors $\pi_1(\theta) \neq \pi_2(\theta)$. If researcher 1 has draws $\theta^{(j)} \sim \pi(\theta \mid Y^T, \pi_1)$, then for any h,

$$\int h(\theta)\, \pi(\theta \mid Y^T, \pi_2)\, d\theta \approx \frac{\sum_{j=1}^{m} h(\theta^{(j)})\, \frac{\pi_2(\theta^{(j)})}{\pi_1(\theta^{(j)})}}{\sum_{j=1}^{m} \frac{\pi_2(\theta^{(j)})}{\pi_1(\theta^{(j)})}}.$$

Derivation for Prior Robustness

$$\int h(\theta)\pi(\theta \mid Y^T, \pi_2)\, d\theta = \frac{\int h(\theta)\, f(Y^T \mid \theta)\pi_2(\theta)\, d\theta}{\int f(Y^T \mid \theta)\pi_2(\theta)\, d\theta} = \frac{\int h(\theta)\, \frac{\pi_2(\theta)}{\pi_1(\theta)}\, \pi(\theta \mid Y^T, \pi_1)\, d\theta}{\int \frac{\pi_2(\theta)}{\pi_1(\theta)}\, \pi(\theta \mid Y^T, \pi_1)\, d\theta}.$$

Importance Sampling Summary

- Choose g close to π (match location, scale, tails).
- Use self-normalized IS if normalizing constants unknown.
- Diagnose with RNE; respecify g if RNE is low.

Takeaways

- Simulation approximates expectations under complex posteriors/marginals.
- RNG quality matters; use proven libraries.
- Acceptance sampling is simple but can be inefficient.
- Importance sampling is powerful; success hinges on a good proposal.