Monte Carlo Methods

Useful refs:

Monte Carlo Stat. Methods : Robert & Casella

MCMC: Stochastic Simulation for Bayesian Inference : Gamerman & Lopes

Ross: Ch. 11

No reqd. text: lecture notes + occasional handouts

Computing assignments : must be typed up.
(easiest in 'R')

Monte Carlo

Basic idea: learn about prob. models by simulating them. Useful for both probability & stat. inference.

Formally: Use pseudo-random (simulated) values from a prob. distr. f to estimate expectations w.r.t. f. Also useful for integration in general.

Suppose we want to find μ where

$$\mu = E_f(g(X)) = \int g(x) f(x)\, dx$$

$$\left(\text{OR } \sum_i g(x_i) f(x_i) \text{ if } f \text{ is pmf}\right)$$

Let $X_1, \ldots, X_n \overset{iid}{\sim} f$ using a computer

Define $\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^{n} g(X_i)$

$\therefore$ $\hat{\mu}_n$ is the Monte Carlo estimate of μ.

Strong Law of Large Numbers:

If $E_f|g(X)| < \infty$ then $\hat{\mu}_n \to \mu$ a.s.

a.s. ('almost sure') convergence: $P\left(\lim_{n\to\infty} \hat{\mu}_n = \mu\right) = 1$

If $E_f|g(x)| < \infty$ and $\sigma^2 = Var_f(g(x)) < \infty$

then by the Central Limit Thm

$$\sqrt{n}\,(\hat{\mu}_n - \mu) \longrightarrow N(0, \sigma^2) \quad \text{in distribution}$$

Convergence in distr. (weak convergence, convergence in law):

$X_n \to X$ in distr. if $F_{X_n}(x) \to F_X(x)$ as

$n \to \infty$ for all x s.t. $F_X(x)$ is cntns.

$F_X(x) = P(X \le x)$, the cdf.

Note: Informally distr. of $\hat{\mu}_n$ approaches $N(\mu, \sigma^2/n)$

so Monte Carlo s.error of $\hat{\mu}_n$ decreases as $1/\sqrt{n}$.

Above results hold for multidimensional spaces (not just 1-D r.v.s).

Accuracy of M.C. estimate is independent of dimensionality of space sampled.

Thinking about iid Monte Carlo = thinking about basic statistics.

Midterm 2011

Mean 17.7

Median 18.5

V. roughly:

[20, 25]: excellent

[16, 20): fine-good

≤ 15: possible cause for concern

Midterm prob.:

10 1. standard M.C. — (a) set-up; (b)–(f) standard stuff (no pts off if error due to (a))

4 2. detailed balance / standard M.C. (done in class)

2 3. (a) basic P.P.

2 (b) condtl P.P. X_1, X_2, X_3 iid Unif$(0, t)$ } more thought

3 (c) memoryless / conditioning for exp. } , but like hw

2 4. (a) iterated expec.

2 (b) using indep. incr. } tricky

Toy example (Geyer's notes)

$X \sim N(0,1)$, $Y \sim N(0,1)$, independent

What is $P(Y < X^2)$?

Let $\mu = P(Y < X^2) = \int \underbrace{\Phi(x^2)}_{P(X < x^2) \text{ for } N(0,1)} \phi(x)\,dx$ pdf for $N(0,1)$

Monte Carlo approach: simulate $(X_1, Y_1), \ldots, (X_n, Y_n)$ of pairs of $N(0,1)$ r.v.'s

$$\hat{\mu}_n = \frac{\sum_{i=1}^{n} I(Y_i < X_i^2)}{n}$$ (sample prop.)

This is easy enough that numerical integration also works, but often numerical integration is not an option.

3. If $\mu = E_f(g(x)) < \infty$, we have:
$\hat{\mu}_n \to \mu$ a.s. (almost surely) by the Strong Law of Large Numbers (S.L.L.N.) That is,

$$P(\lim_{n\to\infty} \hat{\mu}_n = \mu) = 1. \tag{1.1}$$

Furthermore, if $\sigma^2 = \text{Var}_f(g(x)) < \infty$, we can establish a convergence rate for this estimate, that is we can establish how quickly $\hat{\mu}_n$ converges to μ from the **Central Limit Theorem**:

$$\sqrt{n}(\hat{\mu}_n - \mu) \to N(0, \sigma^2) \text{ in distribution} \tag{1.2}$$

Example 1: Suppose we want to calculate $\Pr(-1 < X < 0)$ when X is a Normal(0,1) random variable. We could easily do this by Monte Carlo:

- Generate $X_1, \ldots, X_n \sim N(0, 1)$.
- Compute the estimate

$$\hat{\mu}_n = \frac{\sum_{i=1}^{n} 1(-1 < X_i < 0)}{n},$$

  which is simply the proportion of times $X_i \in (-1, 0)$ for sampled values $X_1, \ldots, X_n$.

For large enough n, $\hat{\mu}_n$ will be very close to $\Pr(-1 < X < 0)$. Of course, Monte Carlo is not really needed for this toy problem since statistical software can easily calculate such probabilities.

start → here

Example 2: Suppose we want to conduct a simple hypothesis test to see if the correlation ρ between two random variables is significant. Assume that the two random variables X and Y come from a bivariate normal distribution, and we observe 30 data points $(X_1, Y_1), \ldots, (X_{30}, Y_{30})$, and want to conduct a hypothesis test based on these data. The sample correlation, $\hat{\rho}$, is the test statistic. For these data, $\hat{\rho} = 0.3$. To find the associated p-value, we need to find the probability $P(\hat{\rho} > 0.3)$ under the null hypothesis that there is no correlation ($\rho = 0$), and the alternative that $\rho > 0$. To calculate this probability, we would need to know the sampling distribution of $\hat{\rho}$ under the null hypothesis that $\rho = 0$. This null distribution is not easy to calculate, but it is given in Anderson (2003). The sample corrrelation coefficient $\hat{\rho}$, for a sample of size N from a bivariate Normal with mean $\boldsymbol{\mu}$ and correlation ρ, depends only on ρ and N (not on $\boldsymbol{\mu}$ or the marginal variances), and is given by:

$$f(\gamma) = \frac{2^{N-3}(1-\rho^2)^{0.5(N-1)}(1-\gamma^2)^{-0.5(N-4)}}{\pi\Gamma(N-2)} \sum_{i=0}^{\infty} \Gamma\left(\frac{N+\alpha-1}{2}\right)^2 \frac{(2\rho\gamma)^\alpha}{\alpha!},$$

where $\gamma \in (-1, 1)$. Now, for $\rho = 0$, the sampling distribution simplifies to:

$$f(\gamma) = \frac{2^{N-3}(1-\gamma^2)^{-0.5(N-4)}}{\pi\Gamma(N-2)} \Gamma\left(\frac{N-1}{2}\right)^2.$$

Finding the above distribution is a non-trivial and time consuming problem, and even though we can assume that we did not have to work to find the distribution (since the theory has already been worked out), finding the p-value for this hypothesis test would still involve integrating the above density over the interval $(0.3, \infty)$. However, if we simply use the fact that the distribution of $\hat{\rho}$ only depends on ρ and the sample size N, a Monte Carlo solution to this problem is very simple:

- To draw a single sample of $\hat{\rho}$ from the null distribution, generate a sample $X_1, \ldots, X_N \sim N(0, 1)$ and a sample $Y_1, \ldots, Y_N \sim N(0, 1)$. (In R you would use the command `xs=rnorm(30,0,1)`, for instance). Find the sample correlation r_1 based on the pairs $(X_1, Y_1), \ldots, (X_N, Y_N)$. Repeat this process m times to obtain m sample correlations $r_1, \ldots, r_m$, generated from the null distribution.

- Compute the estimate

$$\hat{\mu}_n = \frac{\sum_{i=1}^m 1(r_k > 0.3)}{m}.$$

For large enough m, this is an accurate estimate of the desired p-value.

6 4(b)

Note that the only theory necessary was recognizing that the sampling distribution of $\hat{\rho}$ depends only on ρ and N, which is easy to prove by the invariance of the distribution of $\hat{\rho}$ to affine transforms of a bivariate normal random variable (see Anderson (2003) for details). It was not necessary to derive the complicated formula for the sampling distribution above, nor was it necessary to compute the integral; the p-value is easily estimated through a simple Monte Carlo procedure. Of course, since the exact distribution is available here, and the p-value only involves a 1-dimensional integral, it is possible to do this by using numerical integration procedures. In more complicated situations, Monte Carlo will often be the only solution.

1.3 Monte Carlo standard errors

Informally, (1.2) states that the distribution of $\hat{\mu}_n$ approaches a Normal distribution, $\mathrm{N}(\mu, \sigma^2/n)$. Hence, to obtain confidence intervals and error estimates for the estimator $\hat{\mu}_n$, we need to estimate σ^2. Monte Carlo standard error is an assessment of the error of our Monte Carlo estimator. For the simple independent Monte Carlo scenario above, we can easily estimate σ^2 by the sample variance, $\hat{\sigma}^2$. Then, the estimate of Monte Carlo standard error is simply $\hat{\sigma}/\sqrt{n}$. Since $\hat{\sigma}^2$ is a consistent estimator of σ^2, we can use Slutsky's Theorem and (1.2) to obtain asymptotic 95% confidence intervals for Monte Carlo estimates in the usual way: $\hat{\mu}_n \pm 1.96\hat{\sigma}/\sqrt{n}$ ((1.2) still holds when σ^2 is replaced by a consistent estimator of σ^2). We note the following:

1. The independence requirement for the samples is unnecessarily restrictive as we will see later on when we discuss Markov chain Monte Carlo methods. The S.L.L.N. will typically hold under similar conditions for the dependent case but the C.L.T. may not hold and estimating σ^2 is typically very difficult.

2. f may be multivariate, i.e., the random variable for which f is the probability distribution may be multidimensional.

If we can simulate draws from f, easy to estimate expectations w.r.t. f.

Generally hard to do for multivariate random quantities.

Few exceptions: multivariate normal, Wishart.

How do we simulate samples from an arbitrary distr. f? General strategy:

1) Build a method to generate $U_1, U_2, \ldots \overset{iid}{\sim} Unif(0,1)$.
2) Generate $X \sim f$ using $U_1, U_2, \ldots$

i) Uniform pseudo-random number generator: starting from initial value u_0, called the 'seed', produce a <u>deterministic</u> sequence of values in $[0,1]$, $u_1, u_2, \ldots$ by $u_k = d(u_{k-1}, u_{k-2}, \ldots)$ for some function d. This sequence imitates a sequence of iid uniform r.v.s.

end 3/10

This sequence is not random but is an acceptable pseudo-random sequence if it passes a set of tests s.t. the hypothesis

$u_1, \dots, u_n \overset{iid}{\sim} Unif(0,1)$ is reasonable

e.g. Marsaglia's 'Die-Hard' tests.

Typical uniform random # generator: of form:

Initial value : X_0 (seed)

$$X_{n+1} = \underbrace{(a X_n + c) \text{ modulo } m}_{\text{remainder from dividing } aX_n+c \text{ by } m} \qquad a, c, m : \mathbb{Z}^+$$

so $X_n \in \{0, \dots, m-1\}$

X_n/m approximation to $Unif(0,1)$.

Lots of generators, increasingly sophisticated : Knuth's TAOCP , Marsaglia's Super-Duper, Mersenne-Twister etc. see help(Random.seed).

Properties of unif. r.v. generators:

1) Repetitive: after enough draws, pattern repeats itself. # draws before repetition = period.

2) Not independent.

3) Not quite uniform continuous (actually discrete).

Lots of v. good uniform r.v. generators in existing stat. software — we will generally not worry abt. it, (except if using parallel computing. Why: Start M.C. on different machines - results not as indep. as we would like.)

E.g. R uses Mersenne-Twister w/ period $2^{19937}-1$.

Note: advantage of deterministic behavior: reproducibility.

In R: Random Seed : can look at seed.

2) How do we generate $X \sim f$ using uniform r.v.'s?

For important r.v.'s special techniques exist.
See Ross's book for normal, gamma, χ^2, beta, etc.

More general approach: Inverse CDF / inverse transformation method

Let $U \sim \text{Unif}(0,1)$. If F is a cdf
$X = F^{-1}(U)$ has cdf F. $F^{-1}(u) = \{x : F(x) = u\}$

Pf: $\text{Prob}(F^{-1}(U) \le x) = \text{Prob}(U \le F(x))$
$= F(x) \quad \because U \sim \text{Unif}(0,1)$.

Very limited use: only if univariate r.v. and F has explicit inverse. E.g. Cauchy, Exponential, Weibull, Logistic, but not normal, beta, and many gammas.

Drawing $X \sim f$ is generally hard, especially in high dimensions.
Further complication: Often normalizing constants are unknown.

Rejection (Accept-Reject) Sampling

Goal: Simulate $X \sim f(x)$ when we have $h(x)$ s.t. $h(x)/c_1 = f(x)$, i.e. only know $f(x)$ up to a ~~const~~ normalizing constant c_1 (we don't know c_1).

Suppose we have a simpler pdf $g(x)$ s.t. it is easy to draw $Y \sim g(x)$.

and we know $\sup_x \frac{h(x)}{r(x)} < K$ for some $K < \infty$

and we know K; with $r(x)/c_2 = g(x)$

Rejection sampler: for $i = 1, \ldots, N$:

Simulate $Y_1, \ldots, Y_N \overset{iid}{\sim} g$

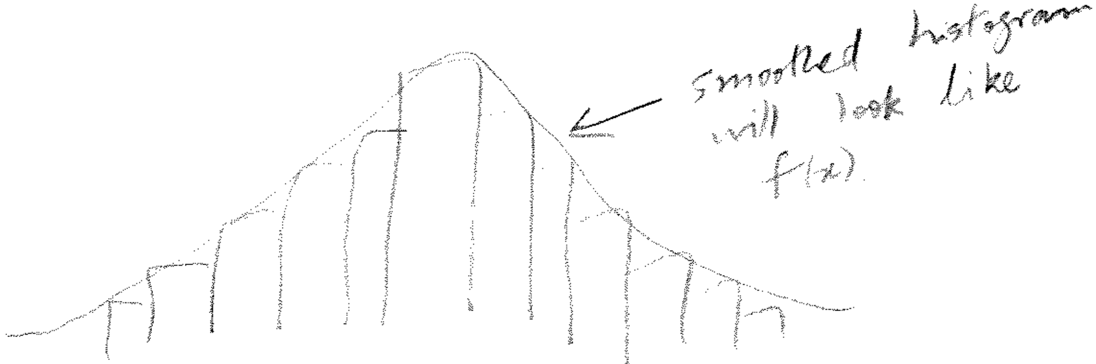
1. Draw $Y_i \sim g$
2. Draw $U \sim \text{Unif}(0,1)$
3. If $U \le \frac{h(Y_i)}{K r(Y_i)}$ return Y_i, else do not.

Values returned by above algorithm are draws from $f(x)$.

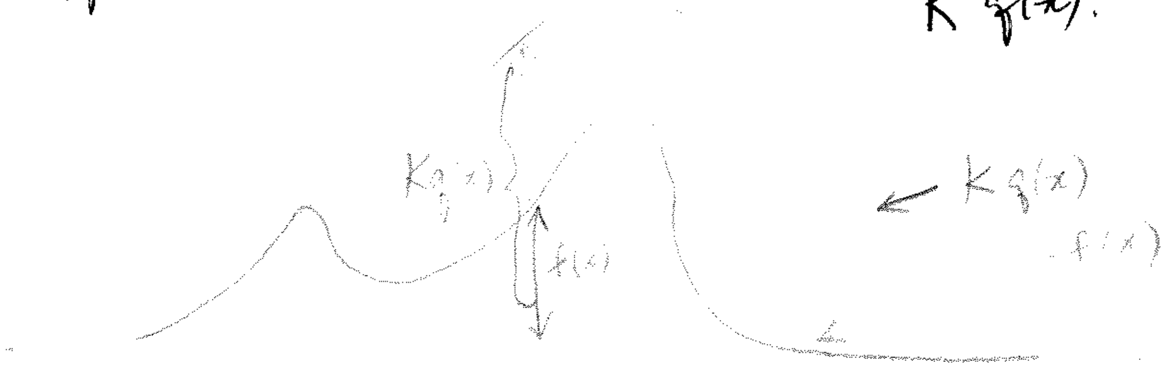
Intuition for rejection sampler

If we could draw directly from $f(x)$.

$X_1, \dots, X_n \overset{iid}{\sim} f(x)$. Large n.



Suppose $f(x)$ is more complicated, but we have envelope $K\,g(x)$.



If we accepted all draws from g would get shape of outer density

But if we ~~reject~~ accept samples w/ prob $\frac{f(x)}{K g(x)}$, can 'recover' inner density.

Proof for rejection sampler (1-D, cntns case).

Let $X \sim q(x)$ and $U \sim \text{Unif}(0,1)$

Accept X if $U \le \frac{h(x)}{K r(x)}$ where $\frac{r(x)}{c_2} = q(x)$, $\frac{h(x)}{c_1} = f(x)$

and $\sup_x \frac{h(x)}{r(x)} \le K < \infty$

Claim: X accepted by this alg. has distr. $f(x)$

Pf: $P(X \le x \mid X \text{ accepted})$

$$= P\left(X \le x \,\middle|\, U \le \frac{h(X)}{K r(X)}\right)$$

$$= P\left(X \le x, U \le \frac{h(X)}{K r(X)}\right) \Big/ P\left(U \le \frac{h(X)}{K r(X)}\right)$$

iterated expectations

$$= \frac{E_q\left\{P\left(X \le x, U \le \frac{h(X)}{K r(X)} \,\middle|\, X\right)\right\}}{E_q\left\{P\left(U \le \frac{h(X)}{K r(X)} \,\middle|\, X\right)\right\}} \overset{\text{condtl. indep.}}{=} \frac{E_q\left\{P(X \le x \mid X) P\left(U \le \frac{h(X)}{K r(X)}\right)\right\}}{E_q\left\{\frac{h(X)}{K r(X)}\right\}}$$

either $X \le x$ or not (given X)

$$= \frac{E_q\left\{I(X \le x) \frac{h(X)}{K r(X)}\right\}}{E_q\left\{\frac{h(X)}{K r(X)}\right\}} = \frac{E_q\left\{I(X \le x) \frac{\cancel{c_1} f(X)}{\cancel{K c_2} q(X)}\right\}}{E_q\left\{\frac{\cancel{c_1} f(X)}{\cancel{K c_2} q(X)}\right\}}$$

uses fact that $\frac{h}{r} < \infty$, hence $\frac{f}{q} < \infty$)

$$= \int_{-\infty}^{x} \frac{f(x)}{q(x)} q(x)\,dx \Big/ \int_{-\infty}^{\infty} \frac{f(x)}{q(x)} q(x)\,dx$$

$(\because \frac{h}{r} < \infty, \frac{f}{q} < \infty)$ $\Rightarrow$ $\frac{P(X \le x)}{}$ $F(x)$, cdf corresponding to $f(x)$. [norm. constant not needed]

Result holds much more generally (multivariate, discrete etc.)

likelihoods

Computing tip #1: when evaluating ratio of pdf's etc.
v. useful to take logs whenever possible

~~For eg.~~ $\cancel{\frac{g(x)\, f(x)}{g(x)} = \exp\{\log g(x) + \log f(x) - \log g(x)\}}$

Fewer problems w/ round-off error, ~~instability of~~
numerical instability.

For eg.
Rej. sampling: Instead of ~~computing~~ evaluating:

Is $\quad u \le \dfrac{f(x)}{K\, g(x)}$?

Try: Is $\log(u) \le \log f(x) - \log g(x) - \log K$?

Computing tip #2: use apply, sapply (~~eg.~~ see help and Rhints).

Tip #3: printing out intermediate values,
debugging: use cat("this var =", ~~varname~~ varname, "\n");
OR print(varname).

COMPUTING TIP: 16(d)

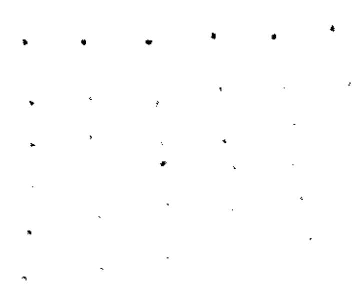
(Please attach additional sheets as needed)

E.g. Where normalizing constant is unknown

E.g.1. Binary Markov Random Field — Look up Ising Model from stat mechanics

N individuals 'pixels' arranged in a lattice. Each individual is in one of 2 state: $+1$ or -1 (eg. black or white)

Ω = space of all configurations, each config $= (s_1, \ldots, s_N)$, where $s_i = +1$ or -1.



Suppose model: $P\left(S = (s_1, \ldots, s_n)\right) = \frac{1}{Z} \exp\{-J\,\mathcal{U}(\underset{\sim}{s})\}$

where $J > 0$, fixed.

And $\mathcal{U}(\underset{\sim}{s}) = \sum_{i \sim j} I(s_i \neq s_j)$ $i \sim j$ if i is a nhbr of j

Z is a normalizing constant.

$Z = \sum_{\underset{\sim}{s} \in \Omega} \exp\{-J\,\mathcal{U}(\underset{\sim}{s})\}$ $\Rightarrow$ have to sum over 2^N possible configurations. $N=10 \Rightarrow 1.27\times 10^3$

$\mathcal{U}(\underset{\sim}{s})$ = # of unlike nhbr pairs in configuration

This model favors smooth images, especially for large J.

We would want a simulation strategy that does not require normalizing constant to be known.

E.g. 2. Simple Bayesian model.

$\mathcal{L}(Y|\Theta)$: likelihood of data Y given parameters Θ.

$p(\Theta)$: prior distr. on Θ.

Inference based on posterior, $\pi(\Theta|Y)$

$$\pi(\Theta|Y) \cancel{=} \frac{\mathcal{L}(Y|\Theta)\, p(\Theta)}{\int \mathcal{L}(Y|\Theta)\, p(\Theta)\, d\Theta} \begin{array}{l} \leftarrow \text{fn. of } \Theta \\ \leftarrow \text{constant w.r.t. } \Theta \end{array}$$

$$\propto \frac{\mathcal{L}(Y|\Theta)\, p(\Theta)}{Z} \leftarrow \text{unknown and potentially complicated especially if lots of parameters.}$$

Very rarely is $\pi(\Theta|Y)$ tractable / available in closed form

Recap. For rejection sampler we need:

1) q s.t. $\sup_x \{h(x)/q(x)\} < \infty$ $\left(\frac{h(x)}{c_1} = f(x)\right)$

q should have heavier tails than h.

2) Value of K s.t. $\left(\frac{r(x)}{c_2} = g(x)\right)$

$$\sup_x \frac{h(x)}{r(x)} < K \quad , \quad K < \infty.$$

For efficient sampler, try to find smallest K satisfying above req.

3) q must match h well or else alg. will be v. inefficient.

4) Must be easy to simulate from q, i.e. need to draw values from q quickly for alg. to be efficient.

end 9/24/07

Note: ① Do not need to know normalizing constant of $f(x)$. In fact, do not need normalizing constant of $g(x)$. ~~Why is this useful?~~

~~If g is multivariate normal, evaluating~~

② Can show acceptance probability $= \frac{c}{K}$