Statistical Computing (36-350)

Lecture 16: Simulation III: Monte Carlo

Cosma Shalizi

21 October 2013

Agenda

- Monte Carlo approximation of integrals and expectations
- The rejection method and importance sampling
- Markov Chain Monte Carlo

READING: Handouts on the class webpage
OPTIONAL READING: Geyer, "Practical Markov Chain Monte Carlo", *Statistical Science* **7** (1992): 473–483;

Random Samples and Integrals

Law of large numbers: if $X_1, X_2, \ldots X_n$ all IID with p.d.f. p,

$$\frac{1}{n}\sum_{i=1}^{n} f(X_i) \rightarrow \mathbb{E}_p[f(X)] = \int f(x)p(x)dx$$

The **Monte Carlo principle**: to find $\int g(x)dx$, draw from p and take the sample mean of $f(x) = g(x)/p(x)$

Examples

Buffon's needle (homework!)

Examples

Buffon's needle (homework!)
Area of a complicated shape C: draw X uniformly from box around C, take average of $\mathbf{1}_C(X)$

Examples

Buffon's needle (homework!)

Area of a complicated shape C: draw X uniformly from box around C, take average of $\mathbf{1}_C(X)$

Any expectation value, variance, ...

Examples

Buffon's needle (homework!)

Area of a complicated shape C: draw X uniformly from box around C, take average of $\mathbf{1}_C(X)$

Any expectation value, variance, …

Anything your other classes teach you as integrals or expectations: significance levels, risk of portfolios, revenue of ads, thresholds for epidemics, …

Bayes's Rule and Integrals

Bayes's rule:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int p(y|x')p(x')dx'}$$

Seems like we need to know the integral

$$p(y) = \int p(y|x')p(x')dx'$$

Monte Carlo can be very accurate

Central limit theorem:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{g(x_i)}{p(x_i)} \rightsquigarrow \mathcal{N}\left(\int g(x)dx, \frac{\sigma^2_{g/p}}{n}\right)$$

Monte Carlo approximation to the integral is unbiased
RMS error $\propto n^{-1/2}$
$\therefore$ Just keep taking Monte Carlo draws, and the error gets as small as you like, even if g or x are very complicated

Generating from p is easy if it's a standard distribution or we have a nice, invertible CDF (quantile method)
What can we do if all we've got is the probability density function p?

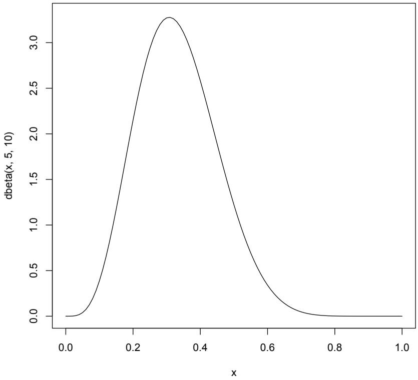
Suppose the pdf f is zero outside an interval $[c,d]$, and $\leq M$ on the interval

Draw the rectangle $[c,d] \times [0,M]$, and the curve f

Area under the curve $= 1$

Area under curve and $x \leq a$ is $F(a)$

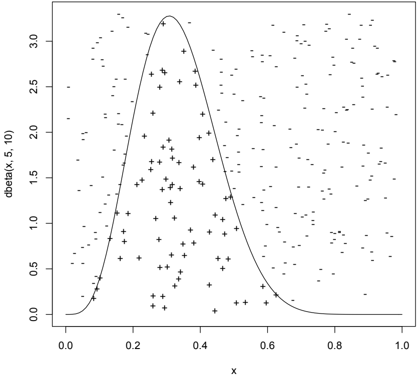
How can we uniformly sample area under the curve?



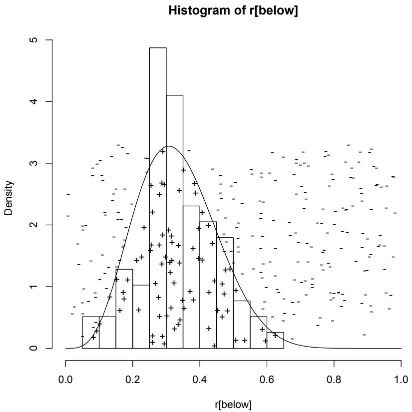

```
M <- 3.3; curve(dbeta(x,5,10),from=0,to=1,ylim=c(0,M))
```

We sample uniformly from the *box*, and take the points under the curve

We sample uniformly from the *box*, and take the points under the curve
$R \sim \text{Unif}(c,d)$
$U \sim \text{Unif}(0,1)$
If $MU \leq f(R)$ then $X = R$, otherwise try again




```
r <- runif(300,min=0,max=1); u <- runif(300,min=0,max=1)
below <- which(M*u <= dbeta(r,5,10))
points(r[below],M*u[below],pch="+"); points(r[-below],M*u[-below],pch="-")
```




```
hist(r[below],xlim=c(0,1),probability=TRUE); curve(dbeta(x,5,10),add=TRUE)
points(r[below],M*u[below],pch="+"); points(r[-below],M*u[-below],pch="-")
```

If f doesn't go to zero outside $[c,d]$, try to find another density ρ where

- ρ also has unlimited support
- $f(a) \leq M\rho(a)$ everywhere
- we can generate from ρ (say by quantiles)

Then $R \sim \rho$, and accept when $MU\rho(R) \leq f(R)$
(Uniformly distributed on the area under ρ)

Need to make multiple “proposals” R for each X
e.g., generated 300 for figure, only accepted 78
Important for efficiency to keep this ratio small
Ideally: keep the proposal distribution close to the target

Importance Sampling

$$\int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx$$

Importance Sampling

$$\int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx$$

$\therefore$ draw $X_1, X_2, \ldots X_n$ IID from q and

$$\frac{1}{n}\sum_{i=1}^{n} f(x_i)\frac{p(x_i)}{q(x_i)} \approx \int f(x)p(x)dx$$

Importance Sampling

$$\int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx$$

$\therefore$ draw $X_1, X_2, \ldots X_n$ IID from q and

$$\frac{1}{n}\sum_{i=1}^{n} f(x_i)\frac{p(x_i)}{q(x_i)} \approx \int f(x)p(x)dx$$

$p(x)/q(x) =$ **importance weights** (ideally close to 1)

How Do We Do Monte Carlo?

Lots of Monte Carlo needs us to sample from an ugly distribution p
Sometimes none of these tricks work well for p
Markov chain Monte Carlo, MCMC: build a Markov chain whose invariant distribution is p
Run the chain, take its values

The Metropolis Algorithm

We know $p(x) = f(x)/c$ but we don't know c
Suppose

$$p(x)q(y|x) = p(y)q(x|y)$$

then p would be invariant ("detailed balance")

$$\frac{q(y|x)}{q(x|y)} = \frac{p(y)}{p(x)} = \frac{f(y)}{f(x)}$$

We don't need to know c!

Metropolis Algorithm (cont'd)

1. Set X_1 however we like, $t \leftarrow 1$
2. Proposal: Draw Z_t from some $r(\cdot|X_t)$
3. Draw $U_t \sim \mathrm{Unif}(0,1)$
4. If $U_t < f(Z_t)/f(X_t)$, then $X_{t+1} \leftarrow Z_t$, else $X_{t+1} \leftarrow X_t$
5. Increase t, go back to 2

Close to, but not quite, rejection method

```
rmetropolis <- function(n,rinitial,rproposal,f) {
  metrostep <- function(x) {
    z <- rproposal(x)
    u <- runif(1)
    return(if(u < f(z)/f(x)) { z } else { x } )
  }
  return(rmarkov(n,rinitial,metrostep))
}
```

Typically, discard first k values (**burn-in**), then only use every m^{th} value (low correlation), or average blocks of length m

but see Geyer's "One Long Run", "Burn-In is Unnecessary", and "On the Bogosity of MCMC Diagnostics"

Sampling from Bayes's Rule

$$p(x|y) \propto p(y|x)p(x)$$

so we can use Metropolis to draw a sample from $p(x|y)$ without really knowing it!
Key to modern Bayesian statistics

Gibbs Sampling

If X has many dimensions s, even writing $f(x) \propto p(x)$ can be hard
Could try to turn $X_1, X_2, \ldots X_s$ into a Markov chain but that might not work
Might be able to get $p(X_i | X_1, \ldots X_{i-1}, X_{i+1}, X_s) = p(X_i | X_{-i})$
The **Gibbs sampler**:

1. Set $X_1, X_2, \ldots X_s$ somehow
2. Pick a random i
3. Update X_i by drawing from $p(X_i | X_{-i})$
4. Go back to (2)

The sampler is a Markov chain on X
The invariant distribution is p

Summary

1. Monte Carlo is a stochastic way of evaluating integrals
   - Or expectation values or probabilities or…
   - Extra useful when the integrand is complicated or the space is high-dimensional
2. Markov chain Monte Carlo approximates integrals as averages over a Markov process with the right invariant distribution