Numerical Methods in Economics
MIT Press, 1998

Notes for Chapter 8: Monte Carlo and Simulation Methods

April 13, 2011

Monte Carlo and Simulation Methods

- Gaussian, monomial, and Newton-Cotes formulas
  - use predetermined nodes
  - aim at high accuracy
  - need many nodes
- Sampling methods
  - Generate a sequence of points
  - Short sequence produces rough approximation
  - Longer sequences produce better approximations
- Monte Carlo sampling methods
  - Use law of large numbers "intuition"
  - Order $N^{-1/2}$ convergence
  - Use probability theory to prove theorems
  - Use number-theoretic methods to generate deterministic sequences which appear random

Monte Carlo Integration

- Probability theory
  - If X_i are i.i.d. r.v.'s, density $q(x)$, and support $[0,1]$, then

$$\bar{X} \equiv \frac{1}{N} \sum_{i=1}^{N} X_i$$

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} X_i = \int_0^1 xq(x)\,dx, \ a.s.$$

$$\text{var}\left(\frac{1}{N} \sum_{i=1}^{N} X_i\right) = \frac{\sigma_x^2}{N}$$

  - If σ_x^2 is not known *a priori*, an unbiased estimator is

$$\hat{\sigma}_x^2 \equiv (N-1)^{-1} \sum_{i=1}^{N} (X_i - \bar{X})^2$$

- LLN suggests Monte Carlo quadrature:
  - If $X \sim U[0,1]$, then
$$I = \int_0^1 f(x)\,dx = E\{f(X)\}$$
  - The *crude Monte Carlo* method makes N draws from $U[0,1]$, $\{x_i\}_{i=1}^N$, and defines
$$\hat{I} \equiv \frac{1}{N}\sum_{i=1}^N f(x_i)$$
$$\hat{\sigma}^2 = (N-1)^{-1}\sum_{i=1}^N \left(f(x_i) - \hat{I}\right)^2$$
- $\hat{I}$ is a statistical estimate of $\int_0^1 f(x)\,dx$
  - $\hat{I}$ is an unbiased estimate of $\int_0^1 f(x)\,dx$
  - The variance of the $\hat{I}$ estimate is
$$\sigma_{\hat{I}}^2 = N^{-1}\int_0^1 (f(x) - I)^2\,dx = N^{-1}\sigma^2$$

Variance Reduction Techniques

- Monte Carlo estimates have high variance; need to reduce variance
- Antithetic Variates
  - Induce negative correlation in $f(x)$ values
  - Form the estimate
$$\hat{I} = \frac{1}{2N} \sum_{i=1}^{N} \left(f(x_i) + f(1 - x_i) \right).$$
  - If f is monotone, I, has smaller variance than crude estimate
- Control Variates
  - Suppose φ is similar to f but easily integrated.
  - The identity $\int f = \int \varphi + \int (f - \varphi)$ reduces the problem to
    - a Monte Carlo integration of $\int (f - \varphi)$
    - plus the known integral $\int \varphi$.
  - If $cov\,(f, \varphi)$ is large, variance is reduced

- Importance Sampling
  - Sample $f(x)$ where its value is most important
  - If $p(x) > 0$, and $\int_0^1 p(x)\,dx = 1$, then

$$I = \int_0^1 f(x)\,dx = \int_0^1 \frac{f(x)}{p(x)}\,p(x)\,dx$$

  - If x_i is drawn with density $p(x)$, then

$$\hat{I} = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)}$$

    is an unbiased estimate of I, and variance of $\hat{I}$ is

$$\sigma^2_{\hat{I}} = \frac{1}{N}\left(\int_0^1 \frac{f(x)^2}{p(x)}\,dx - \left(\int_0^1 f(x)\,dx\right)^2\right).$$

  - If $f(x) > 0$ and $p(x) = f(x)/\int_0^1 f(x)$, then $f(x) = I\ p(x)$ and $\sigma^2_{\hat{I}} = 0$.
  - Add constant B to make $f(x)$ positive
  - Aim: find a $p(x)$ similar to $f(x)$
  - Thin tails problem
    - In $\sigma^2_{\hat{I}}$ formula, key term is $f(x)^2/p(x)$
    - if $f(x)^2/p(x)$ is large when $p(x)$ is small, variance is large.
    - Normal density often has thin tails problem

Pseudorandom Number Generation

- Random numbers are seldom used
  - Possible methods
    - Flipping coins
    - Geiger counters measuring radioactivity
  - Disadvantages
    - Expensive given RA salaries
    - RA's complain about radiation risk; now have legal rights
- Monte Carlo propagandists
  - Use deterministic sequences
  - Act as if they are random sequences

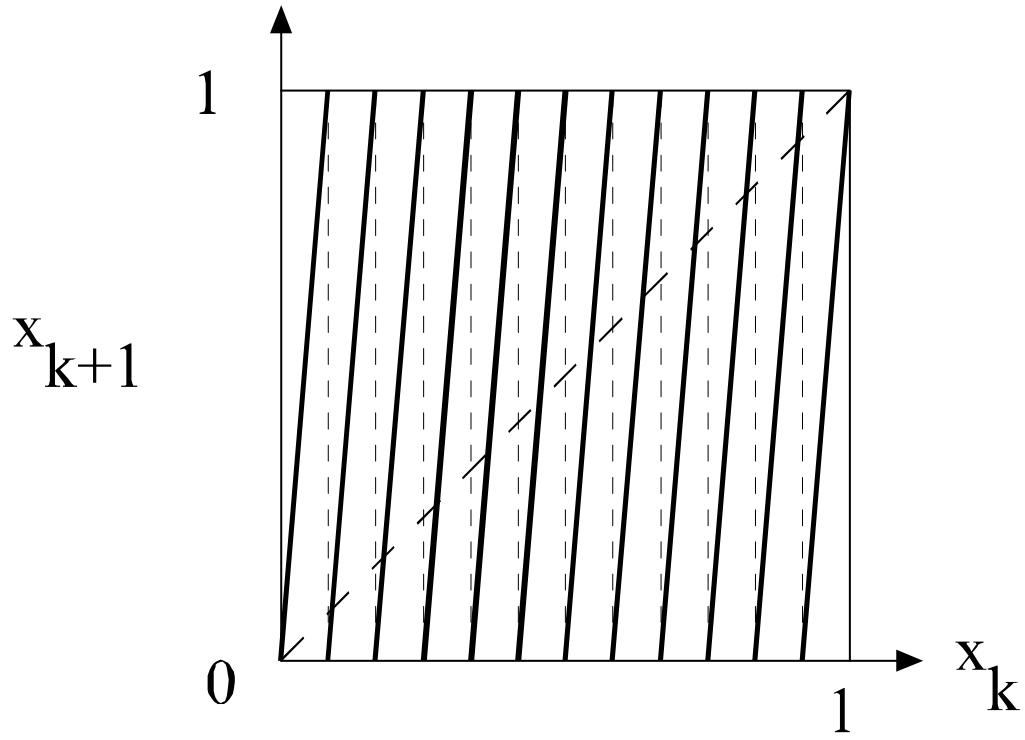
- Pseudorandom numbers are used instead
  - They are deterministic sequences, $X_{k+1} = f(X_k, X_{k-1}, X_{k-2}, ...)$
  - They pass *some* randomness tests, such as
    - Unbiasedness
    $$\frac{1}{N}\sum_{k=1}^{N} X_k = \mu \equiv E\{X\}$$
    - Zero serial correlation
    $$0 = \sum_{k=1}^{N} (X_k - \mu)\, X_{k+1}$$
    - Runs tests
    - Lehmer: "each term is unpredictable to the *uninitiated* and .. digits pass a *certain number* of tests traditional with *statisticians*."
  - They *fail* Brock-Dechert-Scheinkman tests for randomness

Uniform Random Number Generation

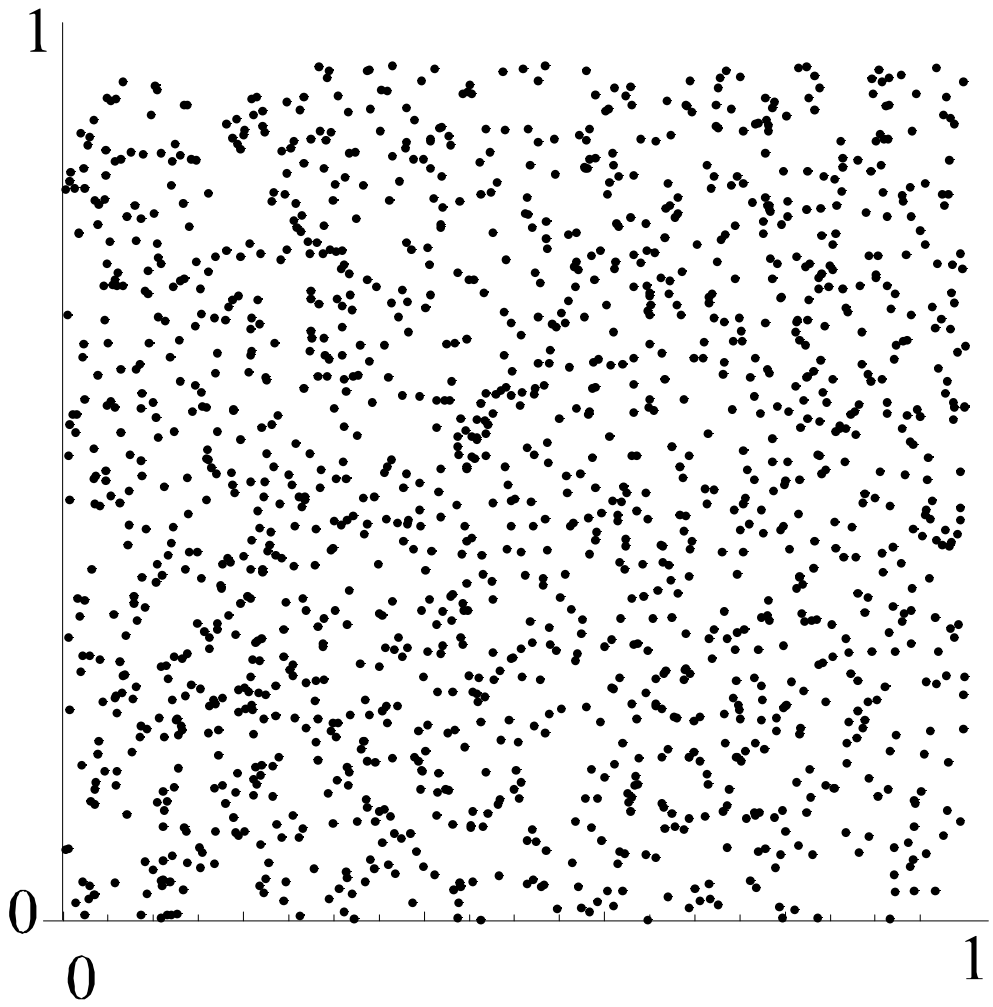
- Linear congruential method (LCM):

$$X_{k+1} = (aX_k + c) \mod m \tag{8.1.1}$$

  - Will generate pseudorandom sequence if parameters chosen well
  - Will eventually cycle; chose parameters to get long cycle
  - $Y_n \equiv (X_{2n+1}, X_{2n+2})$ is a pseudorandom two-dimensional set of points. Similar for R^d
  - LCM generators have fallen into disfavor since they lie on a finite set of hyperplanes.



Linear congruential method function



1500 Points generated by LCM

- Nonlinear schemes:
  - MPRNG: an example of LCM plus "random" shifts.
  - $X_{k+1} = f(X_k) \bmod m$
  - Fibonacci generator $X_k = (X_{k-1} + X_{k-2}) \bmod . m$. This sequence has a number of poor properties. In particular, if X_{k-1} and X_{k-2} are small relative to m, so will be X_k.
  - The Fibonacci-like scheme

$$X_k = (X_{k-24} \cdot X_{k-55}) \; mod 2^{32} \tag{8.1.2}$$

    has a period $\sim 10^{25}$ and passes many randomness tests.

Nonuniform Random Number Generation

- Need to generate nonuniform random numbers
- Inversion:
  - Suppose X has distribution $F(x)$
  - Then $F^{-1}(U)$ has distribution $F(x)$
  - To approximate X, we compute $y_k = F^{-1}(x_k)$ where x_k is a uniform pseudorandom sequence
- Normal random variables: A special method
  - Suppose U_1 and $U_2 \sim U[0,1]$
  - Then X_1, $X_2 \sim N(0,1)$ are independent where

$$\begin{aligned} X_1 &= \cos(2\pi U_1)\sqrt{-2\ln U_2}\,, \\ X_2 &= \sin(2\pi U_1)\sqrt{-2\ln U_2}\,, \end{aligned} \tag{8.1.3}$$

Mathematical Justification of Using pseudo-Random Numbers

- Probability theory produces theorems about Monte Carlo methods
- Probability theorems do not apply to deterministic sequences such as pseudo-Monte Carlo sequences
- What is the connection?
  - Probability theory tells us that there are many sequences of numbers that will have certain statistical properties and successfully integrate functions
  - The existence of such sequences encourages us to search for them
  - The literature on random number generators engages in this search, and has found several sequences that do well for a finite initial segment
- The only proof of the validity of a random generator is the computational properties we learn from computing it.
- This is not theorem proving; it is more like experimental mathematics!!!

Stochastic Approximation

- Consider

$$\min_x E_Z\{g(x, Z)\}, \tag{8.4.1}$$

  where Z is a random variable.

- Conventional methods are impractical for empirical problems, such as

$$\min_\beta E_Z\{g(\beta, X, Z)\}$$

  where β are parameters, X is data, and Z is random

  – Too costly - curse of dimensionality

  – High-accuracy methods are not necessary in empirical problems since X data is noisy.

- Econometricians frequently fix S, and minimize $\sum_{z\in S} g(\beta, X, z)$.

- *Stochastic approximation* is designed to deal with such problems.

  – Begin with initial guess x^1.

  – Draw z^1

  – $g_x(x^1, z^1)$ is an unbiased estimate of the gradient $f_x(x^1)$

  – Steepest descent method would change guess by $-\lambda_1 f_x(x^1)$ for some $\lambda_1 > 0$.

  – The *stochastic gradient method* executes the iteration

$$x^{k+1} = x^k - \lambda_k g_x(x^k,\ z^k), \tag{8.4.2}$$

    where $\{z^k\}$ is a sequence of i.i.d. draws from Z and λ_k is a changing step size.

Theorem 1 *Suppose that f is C^2. If $\lambda_k \to 0$, $\sum_{k=1}^{\infty} \lambda_k = \infty$, and $\sum_{k=1}^{\infty} \lambda_k^2 < \infty$, then the sequence x^k generated by the stochastic gradient method, (8.4.2), confinded to U will almost surely have a subsequence that converges to a point either on ∂U or at a (possibly local) minimum of f.*

- Example:
  - $\min_{x\in[0,1]} E\{(Z-x)^2\}$, $Z \sim U[0,1]$, with solution $x = 0.5$
  - Let $\lambda_k = 1/k$
  - (8.4.2) becomes

$$x_{k+1} = x_k + \frac{2}{k}(z_k - x_k), \tag{8.4.3}$$

Table 8.1: Statistics of (4.3) for 25 Runs

Iterate	Average x_k	Standard Deviation
1	.375	.298
10	.508	.143
100	.487	.029
200	.499	.026
500	.496	.144
1000	.501	.010

Standard Optimization Methods with Simulated Objectives

- Consider optimization problem for some random variable Z:

$$\min_{x \in U} \; E\{g(x, Z)\} = f(x) \tag{8.5.1}$$

  - For many problems, the objective $f(x)$ and its derivatives can be computed only with nontrivial error.
  - When solving problems of the form (8.5.1) we need to determine how well we need to approximate the integral.
  - We next consider optimization approaches that use simulation ideas.

- Idea: take a sample of Z of size N, and replace $E\{g(x, Z)\}$ in (8.5.1) with its sample mean $\frac{1}{N}\sum_{i=1}^{N} g(x, Z_i)$.

- For example, suppose that we want to solve

$$\min_{x \in [0,1]} E\{(Z - x)^2\}, \tag{8.5.2}$$

  where $Z \sim U[0, 1]$.

- To solve (8.5.2), we take, say, three draws from $U[0, 1]$; suppose they are 0.10, 0.73, and 0.49. We then minimize the sample average of $(Z - x)^2$,

$$\min_{x \in [0,1]} \frac{1}{3}((0.10 - x)^2 + (0.73 - x)^2 + (0.49 - x)^2). \tag{8.5.3}$$

  The solution to (8.5.3) is 0.43, a rough approximation of the true solution to (8.5.2) of 0.5.

- Simple portfolio problem. $u(c) = -e^{-c}$; safe return is $R = 1.01$; risky asset has return $Z \sim N(\mu, \sigma^2)$ with $\mu = 1.06$ and $\sigma^2 = 0.04$. Portfolio problem is

$$\max_{\omega} \; -E\{e^{-((1-\omega)R+\omega Z)}\}. \tag{8.5.4}$$

- Optimal ω, denoted ω^*, and equals 1.25.

- Stochastic optimization uses Monte Carlo integration to evaluate the integral objective.

  – Take N draws of $Z \sim N(\mu, \sigma^2)$, and replace (8.5.4) by

$$\max_{\omega} -\frac{1}{N}\sum_{i=1}^{N} e^{-((1-\omega)R+\omega Z_i)}. \tag{8.5.5}$$

  – Solution to (8.5.5) is $\hat{\omega}^*$; errors depend on N

Table 8.2: Portfolio Choice via Monte Carlo

	$N^{-1}\sum_{i=1}^{N} u(c_i)$		$\widehat{\omega}^*$	
N	mean	Standard deviation	mean	Standard deviation
100	-1.039440	.021362	1.2496	.4885
1000	-1.042647	.007995	1.2870	.1714
10,000	-1.041274	.002582	1.2505	.0536

- Note: error in computing ω^* is ten to twenty times larger than error in computing an expectation.

Estimating Quality of MC procedures

- Suppose that you have a function

$$y = f(x)$$

which is computed by a procedure which of the form

$$Y = F(x, Z)$$

where

  - x are inputs (such as data)
  - y is the true answer given inputs x
  - Z is a set of random draws
  - The variance of the error, $Y - y$, goes to zero like $N^{-1/2}$ as the N, cardinality of Z, goes to infinity

- This is structure of many estimation procedures and stochastic programming methods.

- Question: How can we tell if the random variable Y is a good estimate of the the true y?
- Answer: Use Monte Carlo to estimate the mean and variance of the error $Y - y$.
  - Choose a number of repititions, K.
  - Construct K independent sets Z, Z_i, $i = 1, ..., K$
  - Compute $Y_i = F(x, Z_i)$, $i = 1, ..., K$
  - Compute sample mean and variance of the set $\{Y_i | i = 1, ..., K\}$
- Econometrics application
  - Need to distinguish between sample noise and noise from MC errors.
  - The standard error measures only sample noise, and is a r.v. if you use MC methods; therefore, you need to also compute the variance of the estimate of the standard error!!
  - Why is this not done? Because it makes "sample fishing" more difficult?