Advanced Statistical Computing

Fall 2012
Lecture 1

Steve Qin

Instructors

Hao Wu

Yijuan Hu

Tianwei Yu

Steve Qin

What is this class about?

- Survey the most used computational techniques crucial in modern biostatistics research.
- Discuss practical programming skills that will save your time, effort and even your career.
- Prepare students to take on RA projects.

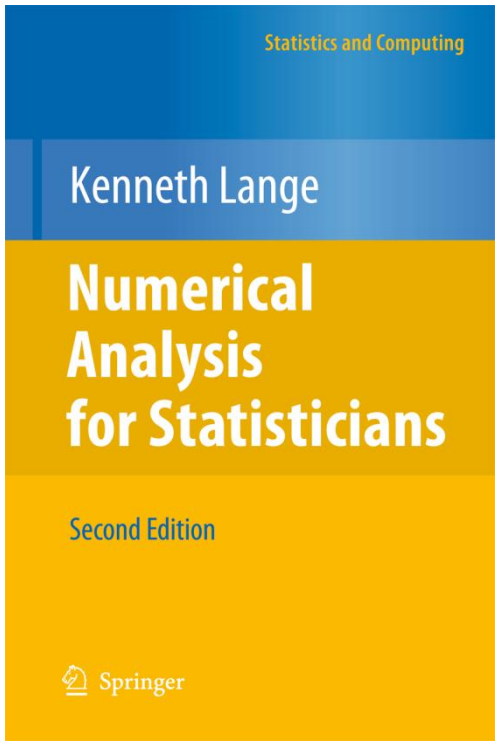
Outline

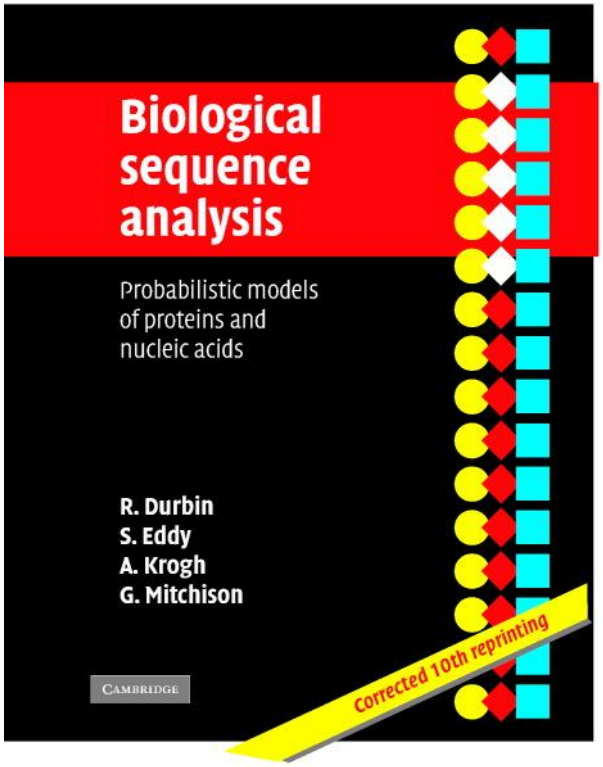
- **Week 1-2**: MCMC (Qin)
- **Week 3-4**: EM algorithm and extensions (Hu)
- **Week 5**: HMM and inference (Yu)
- **Week 6-7**: Linear Programming (Wu)

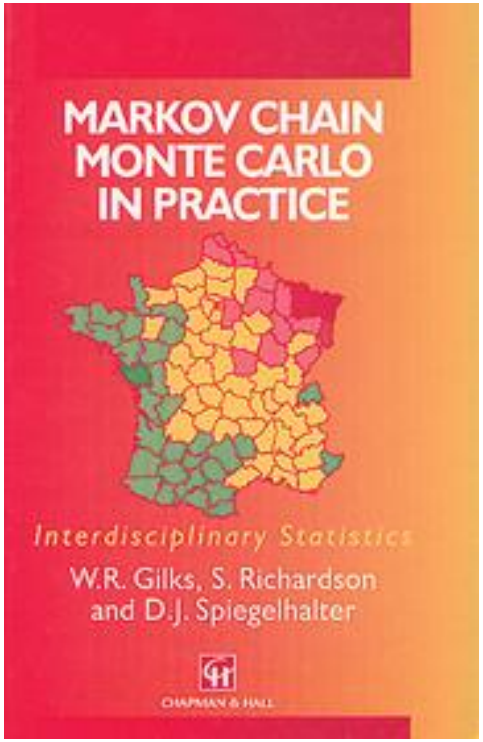
Evaluation

- Four sets of homework
- Requires real programming
- Each worth 25% of the final grade

Reference books







Class website

http://www.sph.emory.edu/~hwu/teaching/statcomp/statcomp.html

In the first 2 weeks

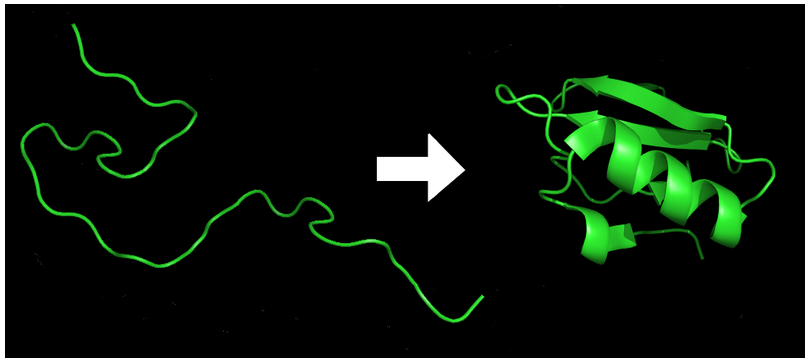
Today

- Examples of computation problems in science and engineering,
- Introduction to Monte Carlo strategy and methods,
- Random number generation,
- Importance sampling.

In the first 2 weeks

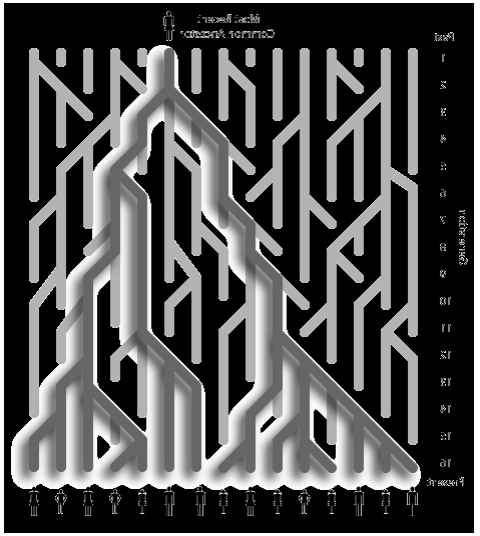
- 9/4
  - Introduction to data augmentation, MCMC idea,
  - Gibbs Sampler,
  - Metropolis-Hastings algorithms.
- 9/6.
  - Check for convergence,
  - Techniques to accelerate Markov chain mixing.
- 9/11.
  - Implementations of Monte Carlo methods,
  - Examples of MCMC applications.

Examples

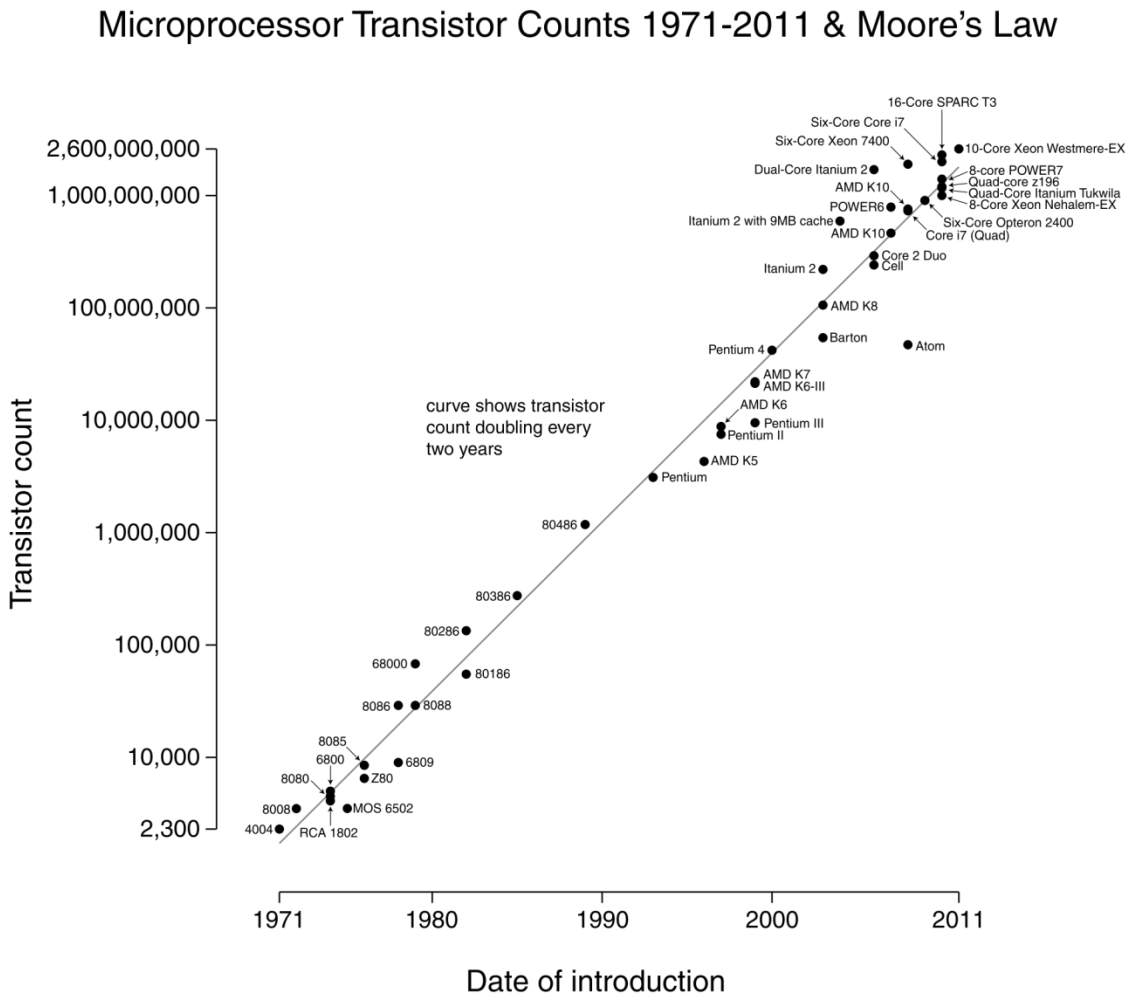


- How protein fold molecular dynamics
- Phylogeny, inheritance pattern in large pedigree
- Next generation sequencing

  mapping, assembly, …



Moore's law

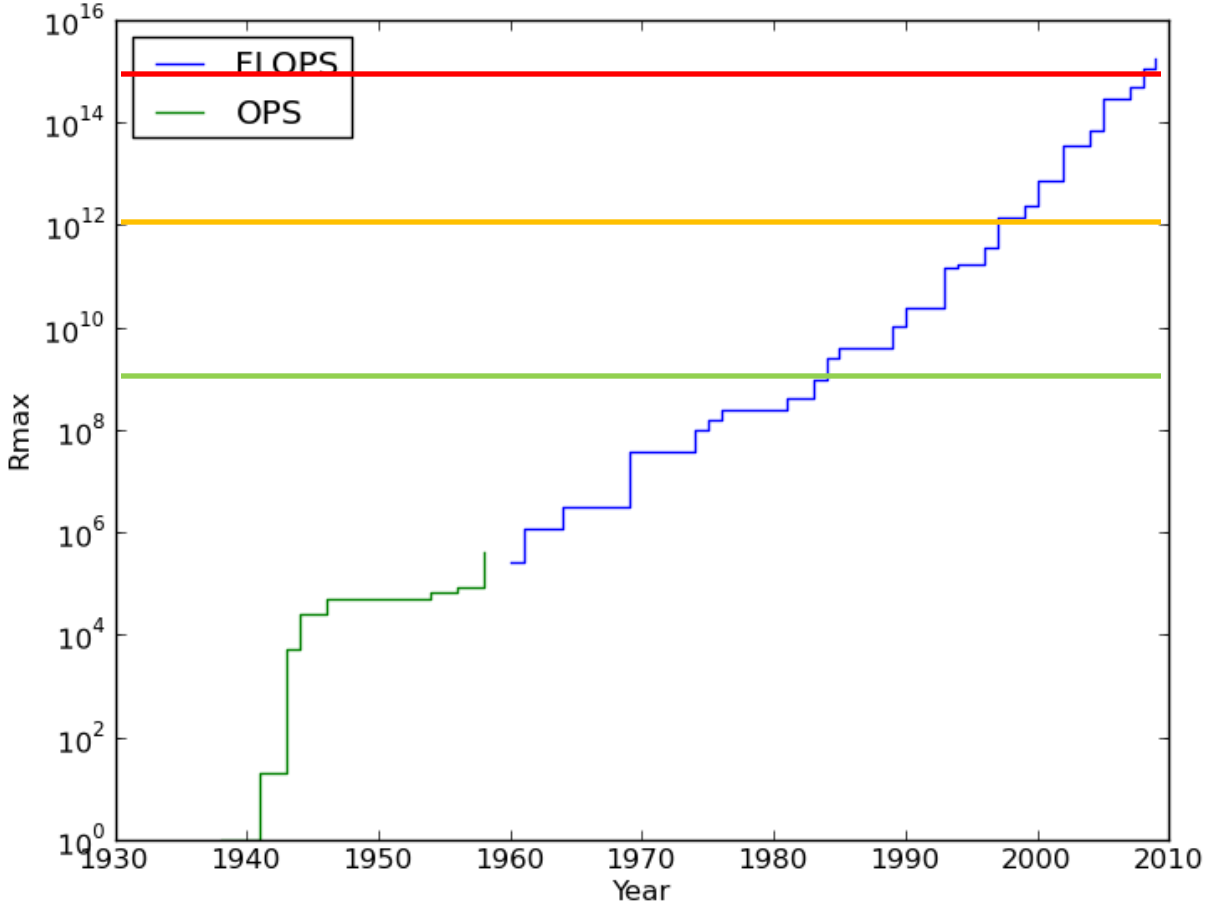

Microprocessor Transistor Counts 1971-2011 & Moore's Law
Transistor count
2,600,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
2,300
curve shows transistor count doubling every two years
16-Core SPARC T3
Six-Core Core i7
Six-Core Xeon 7400
10-Core Xeon Westmere-EX
Dual-Core Itanium 2
8-core POWER7
Quad-core z196
Quad-Core Itanium Tukwila
AMD K10
8-Core Xeon Nehalem-EX
POWER6
Six-Core Opteron 2400
Itanium 2 with 9MB cache
Core i7 (Quad)
AMD K10
Core 2 Duo
Cell
Itanium 2
AMD K8
Barton
Atom
Pentium 4
AMD K7
AMD K6-III
AMD K6
Pentium III
Pentium II
AMD K5
Pentium
80486
80386
80286
68000
80186
8086
8088
8085
6800
6809
8080
Z80
8008
MOS 6502
4004
RCA 1802
1971
1980
1990
2000
2011
Date of introduction

Computation speed

Peta

Tera

Giga



Computation speed

- Intel Core i750 — 7 GFlops.
- The Intel Core2Quad — 6.9 GFlops.
- AMD Phenom II X4 — 7.5 Gflops.
- Core2Dup E8200 — 2.9 GFlops.
  Celeron M 540 — 0.9 GFlops.
- Pentium4 — 0.74 GFlops.
- nVidia GeForce 8800GS — 264 Gflops
- nVidia GeForce 9800GT — 336 GFlops.
  (Results from MaxxPi^2)

What is Monte Carlo?

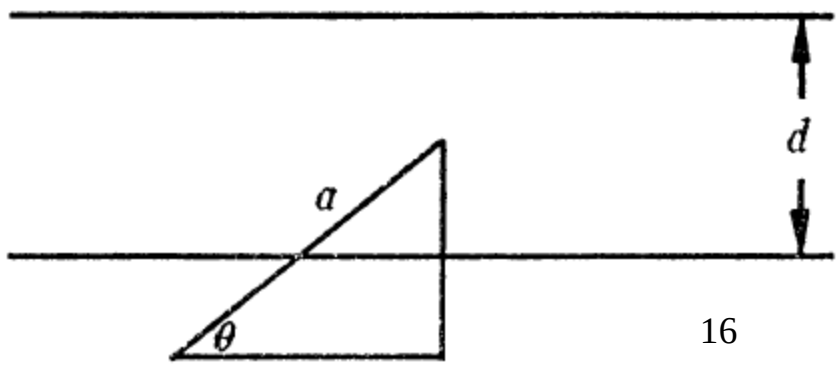
- Rely on repeated sampling to study the results of a experiment or study the properties of certain procedure.
  - Often used in complex and uncertain scenarios
  - Difficult to formulate, high correlation.
  - Cheap
  - Take advantage of faster computers
- History
  - John von Neumann, Stanislaw Ulam, Nicholas Metropolis

Buffon's needle

Georges-Louis Leclerc, Comte de Buffon

(1707-1788)

Given a needle of length *a* and an infinite grid of parallel lines with common distance *d* between them, what is the probability *P*(*E*) that a needle, tossed at the grid randomly, will cross one of the parallel lines?



Buffon's needle

- Assume $a < d$

$$P(E) = \int_0^{\pi} \frac{a \sin\theta d\theta}{\pi d} = (a/\pi d) \int_0^{\pi} \sin\theta d\theta = 2a/\pi d.$$

http://web.student.tuwien.ac.at/~e9527412/buffon.html

Random variate generation

- Generate uniform r.v.

  Uniform (0,1)
- Important techniques
  - Direct method
  - Inverse method
  - Relationship to other distributions
  - Accept-reject method

Direct method

Directly use the definition of the distribution.

- Bernoulli distribution *Bernoulli*(p)

  Generate $r \sim Uniform\ (0,1)$

- Binomial distribution *Binom*(*n*,*p*)

  Generate $r_1,\ldots,r_n$ $iid \sim Uniform\ (0,1)$

Inverse method

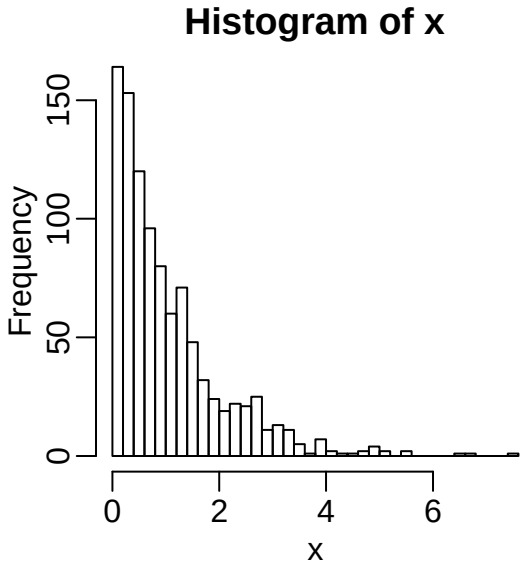
- Theorem: If $U \sim Uniform\ (0,1)$, then using $X = F^{-1}(U)$ generates a random number X from a continuous distribution with specified cdf F.
  - Exponential: generate $r \sim Uniform\ (0,1)$,

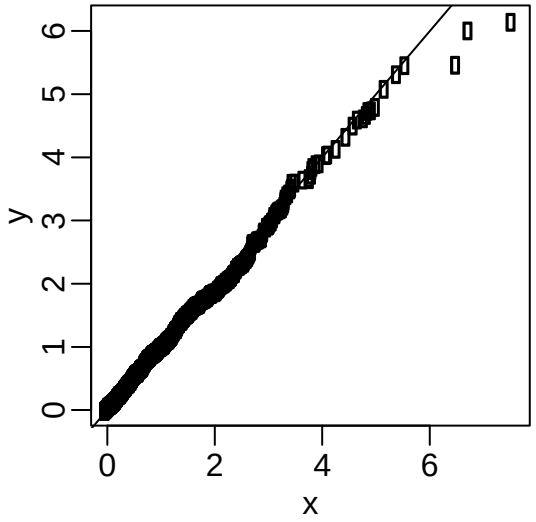
    $F(x) = 1 - e^{-x}$

    then $-\log(1 - r)$ or $-\log(r) \sim Exponential(1)$

Example

```
r=runif(1000) ## generate 1000 U(0,1) random numbers
x=-log(1-r) ## convert to exp(1) random numbers
## compare with exp(1)
y=rexp(1000)
qqplot(x,y); abline(0,1)
```





Numerical approximation

- Is Monte Carlo a “exact” method?
- Use numerical method to approximate complex functions, then use in inverse method to simulate random variates.
- Example: Abramowitz and Stegun 1964 provided numerical approximation to the normal cdf.
  - Error order 10^{-8}

Transformation method (I)

- If a distribution *f* is linked to another distribution *g* which is easy to simulate from.
- Examples: if $X_i's \sim iid\ Exp(1)$

  then

$$Y = 2\sum_{j=1}^{\nu} X_j \sim \chi^2_{2\nu},$$

$$Y = \beta\sum_{j=1}^{a} X_j \sim Gamma(a,\beta),$$

$$Y = \frac{\sum_{j=1}^{a} X_j}{\sum_{j=1}^{a+b} X_j} \sim Beta(a,b).$$

Transformation method (II)

- More examples:
  - Normal distribution: Box-Muller

    generate U_1 , $U_2 \sim iid\ Uniform(0,1)$,

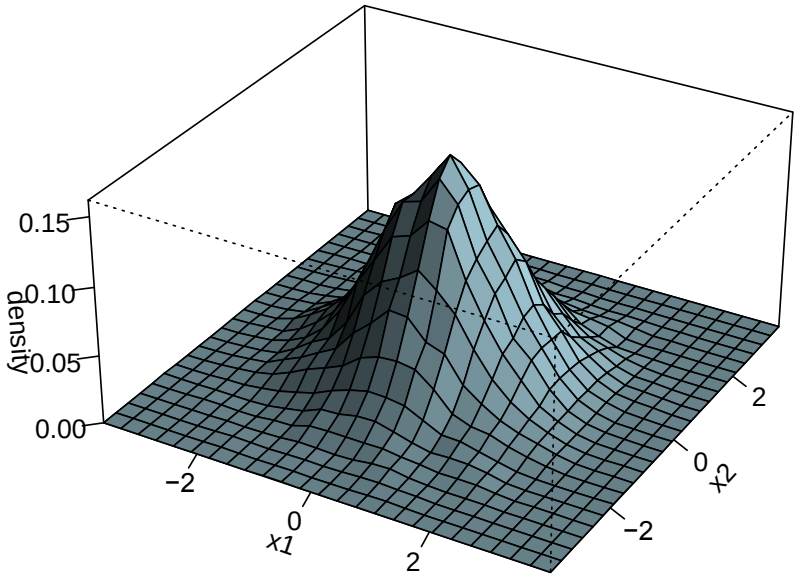
    define

$$\begin{cases} x_1 = \sqrt{-2\log(u_1)}\cos(2\pi u_2), \\ x_2 = \sqrt{-2\log(u_1)}\sin(2\pi u_2). \end{cases}$$

    then x_1 , $x_2 \sim iid\ N(0,1)$.

Example

```
u1=runif(10000); u2=runif(10000)
x1=sqrt(-2*(log(u1)))*cos(2*pi*u2)
x2=sqrt(-2*(log(u1)))*sin(2*pi*u2)
library(MASS)
a=kde2d(x1,x2)
persp(a, xlab="x1", ylab="x2", zlab="density", phi = 30,
        theta = 30, d = 5, shade = 0.75, col = "lightblue",
        expand = 0.5, ltheta = 120,ticktype = "detailed")
```



Transformation method (III)

- Discrete random variables
  - To generate $X \sim P_\theta$, calculate

    $p_0 = P_\theta(X \leq 0)$, $p_1 = P_\theta(X \leq 1)$, $p_2 = P_\theta(X \leq 2)$, …

    then generate $U \sim Uniform(0,1)$ and take

    $X = k$ if $p_{k-1} < U < p_k$.
- *Beta*
  - Generate from Uniform then use order statistics.
- *Gamma*
  - From Beta and Exponential.

Fundamental theorem of simulation

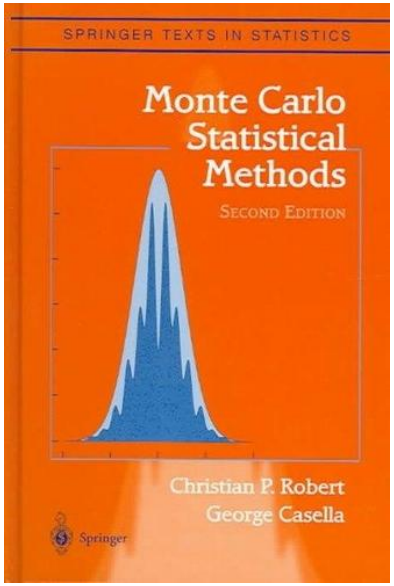
- Simulating $X \sim f(x)$ is equivalent to simulating

  $(X,U) \sim Uniform\{(x,u): 0 < u < f(x)\}$.

  f is the marginal density of the joint distribution.

Accept-reject method

- The accept-reject method
  1. Generate $X \sim g$, $U \sim Uniform(0,1)$,
  2. Accept $Y = X$ *if* $U \leq f(X)/Mg(X)$,
  3. Repeat.



Accept-reject method example

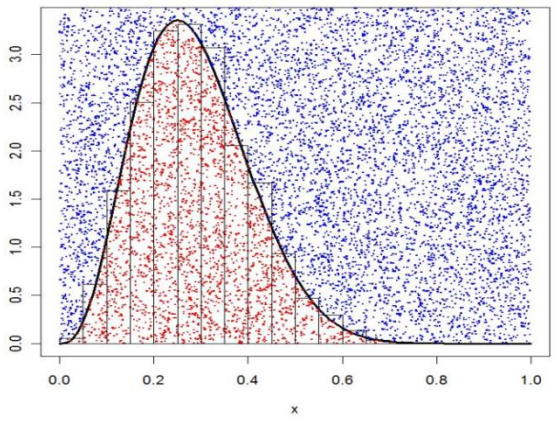
- *Beta* (α,β), $\alpha \geq 1$, $\beta \geq 1$,

  simulate $Y \sim Uniform(0,1)$ and

  $$U \sim Uniform(0,m),$$

  m is the max of the Beta density.

  select $X = Y$ if under curve



what is the acceptance rate?

Importance sampling

- *Importance sampling:*

  to evaluate $E_f[h(X)] = \int_{\aleph} h(x) f(x) dx$

  based on generating a sample $X_1, \cdots, X_n$ from a given distribution g and approximating

$$E_f[h(X)] \approx \frac{1}{m} \sum_{j=1}^{m} \frac{f(X_j)}{g(X_j)} h(X_j)$$

  which is based on

$$E_f[h(X)] = \int_{\aleph} h(x) \frac{f(x)}{g(x)} g(x) dx$$

Importance sampling example (I)

- Small tail probabilities:

  $Z \sim N(0,1)$, $P(Z > 4.5)$

  naïve: simulate $Z_i \sim N(0,1)$, $i=1,\ldots,M$.

  calculate

$$P(Z > 4.5) \approx \frac{1}{M} \sum_{i=1}^{M} I(Z_i > 4.5)$$

Importance sampling example (II)

Let $Y \sim TExp(4.5,1)$ with density

$$f_Y(y) = e^{-(x-4.5)} / \int_{4.5}^{\infty} e^{-x} dx.$$

Now simulated from f_Y and use importance sampling, we obtain

$$P(Z > 4.5) \approx \frac{1}{M} \sum_{i=1}^{M} \frac{\varphi(Y_i)}{f_Y(Y_i)} I(Y_i > 4.5) = .000003377.$$

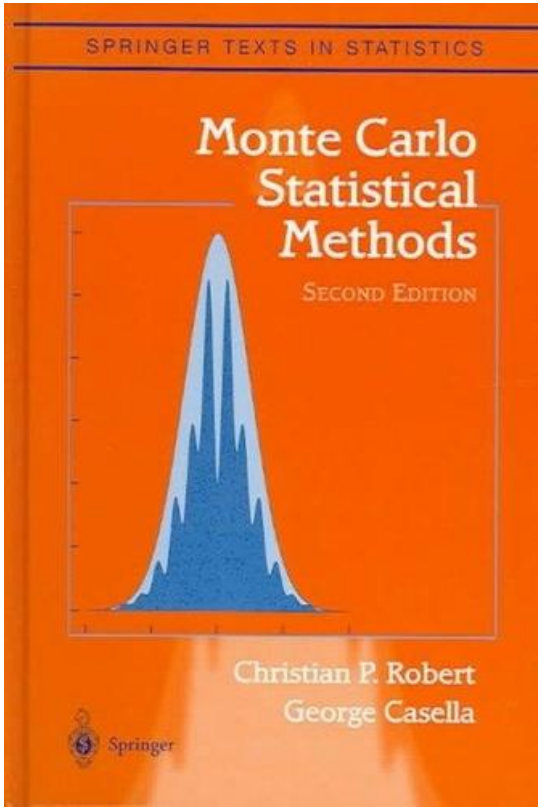
Importance sampling example

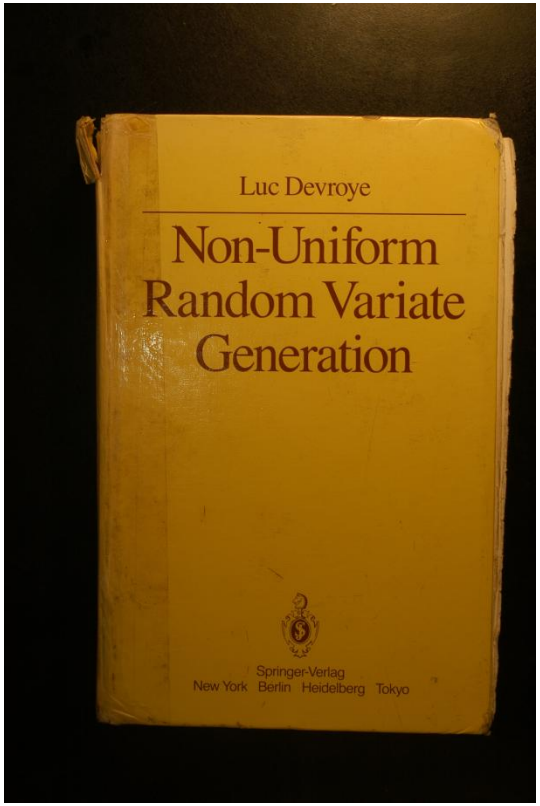
```
## theoretical value
p0=1-pnorm(4.5)
## sample directly from normal distribution
## this needs large number of samples
z=rnorm(10000000)
p1=mean(z>4.5)

## importance sampling
n0=10000
Y=rexp(n0, 1)+4.5
a=dnorm(Y)/dexp(Y-4.5)
p2=mean(a[Y>4.5])

c(p0, p1, p2) ##
[1] 3.397673e-06 2.600000e-06 3.418534e-06
```

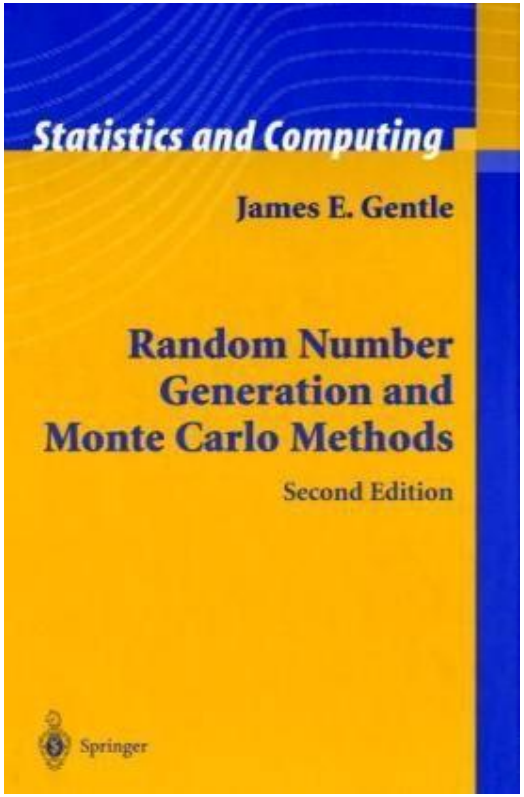
Programming

- In R
  - Lots of functions: runif, rnorm, rbeta, …
- In C/C++
  - Use numerical recipe,
  - Use NAG,
  - Other libraries.

Additional references







Online resources

- Numerical recipe
  http://www.nr.com/
- Luc Devroye's website
  http://luc.devroye.org/rng.html
- Luc Devroye's book
  http://luc.devroye.org/rnbookindex.html