M3/M4S9 Stochastic Simulation

James S. Martin

Department of Mathematics

Imperial College London

Spring 2016

Over the last 50 years, computational techniques have become increasingly important in statistics. Simulation of stochastic systems may be used, for example, to *evaluate* new statistical methodology. Further, it has become recognised that sophisticated simulation methods can be used as the *basis* of new techniques, in statistics, as well as in problems that arise in optimisation and operations research.

This course provides an account of such simulation methods, and covers areas from basic techniques of random variate generation, to modern computational techniques in statistics, such as Markov chain Monte Carlo (MCMC) methods, and bootstrap techniques. Material from M2S1 forms a firm foundation.

The course will consist of two (roughly equal) elements. The first will consist of simulation techniques, and the second will focus on the use of these methods in statistical inference.

The starting point of our discussion, and the basis for everything we cover, is the generation of **pseudo-random numbers** (a deterministic sequence of numbers which has the characteristics of a stream of random variables, uniformly distributed on $[0, 1]$). Then the key objective becomes that of converting these, in an efficient manner, into **random variables from general distributions**: various slick ideas are used. In the first bit of the course we will discuss also ideas of **Monte Carlo integration**, and of the various trick ideas ('computer swindles') that can be utilised with the objective of **variance reduction**, getting the same computational accuracy from a smaller simulation.

The key uses of simulation in statistical inference that we will discuss are: **Monte Carlo tests, MCMC techniques in Bayesian inference** and **bootstrap inference.**

Monte Carlo tests use simulation to build up the distribution of a test statistic under some null hypothesis being tested on data (so replacing the need to 'look up tables'). MCMC techniques are based on the idea that we can sample from some probability distribution of interest (such as the *posterior* distribution in

a Bayesian context) by the (strange looking) device of constructing, then simulating, a Markov chain which has the desired distribution as its *equilibrium distribution*. [We will describe all the necessary elements of Markov chain theory we need]. In bootstrap inference, data from some unknown probability distribution are used to construct an *empirical sampling model*: statistical inference is then performed on the basis of samples simulated from the empirical model. If time permits, we will also consider uses of simulation in particular optimisation problems.

There will be one assessed project, accounting for $\sim 25\%$ of the available course credit. The remaining $\sim 75\%$ will be allocated to a final examination in the summer term. For M3 students, this will be a 1.5-hour exam consisting of three questions; M4 students will sit a 2-hour exam consisting of four questions.

1 Generation of Uniform Random Variates

As the basis for the generation of random variates from particular distributions, e.g. normal, exponential, we shall first study methods of generating from the uniform distribution.

The methods we shall describe generate pseudo-random numbers.

Definition

A sequence of PSEUDO–RANDOM numbers $U_1, U_2, U_3, \ldots$ is a deterministic sequence of numbers in $[0, 1]$ having the same relevant statistical properties of a sequence of random numbers.

PSEUDO - false, apparent, supposed but not real.

Why use pseudo-random numbers rather than real random numbers?

(i) easier, quicker, cheaper.

(ii) repeatable.

There are both good and bad ways of generating pseudo-random numbers, as will be demonstrated in Exercises 1.

1.1 Types of pseudo-random number generators

1.1.1 Congruential pseudo-random number generators

Consider the recursion

$$X_{n+1} \equiv (aX_n + b) \bmod (m)$$

where

X_0	seed (specified),
m	modulus,
b	shift,
a	multiplier.

If $b = 0$ then $X_{n+1} \equiv aX_n \bmod(m)$ $\leftarrow$ **MULTIPLICATIVE GENERATOR.**

If $b \neq 0$ $\leftarrow$ **MIXED GENERATOR.**

We will take $X_0, a, b \in \{0, 1, \ldots, m-1\}$.

This recursion defines a class of "generators". Note that $X_n \in \{0, 1, \ldots, m-1\}$. Now we can generate:

$$U_n = \frac{X_n}{m} \in [0, 1) \ \forall \ n$$

Idea: If m is very large can we choose X_0, m, a, b to give U_n's which "look like" $U(0,1)$'s?

1.1.2 The middle-square method (von Neumann, 1951))

Start with some 4-digit number. e.g.

$$8653$$
$$8353^2 = 74\underbrace{8744}09$$
$$8744$$
$$8744^2 = 76\underbrace{4575}36$$
$$4575$$

The properties of this method are covered in Exercises 1.

1.1.3 Additive Lagged Fibonacci Generators

Consider the recursion,

$$X_{n+1} \equiv (X_n + X_{n-1}) \text{mod}(m)$$

where now X_0, X_1 are the seed and m is the modulus.

Neither the middle-square method nor the lagged Fibonnaci methods are to be recommended; we shall concentrate on congruential generators.

1.2 Properties of congruential generators

1. Given $X_0, X_1, \ldots, X_n$, future values are entirely determined by X_n.

2. $X_0, X_1, \ldots, X_m$ cannot be distinct.

3. $\exists\, i, k \in \{0, 1, \ldots m\}$ s.t. $X_i = X_{i+k}$.

4. $X_i, X_{i+1}, \ldots, X_{i+k-1}$ repeats.

A congruential generator gives a periodic sequence with **PERIOD** k.

- For a mixed congruential generator $k \leq m$.
- For a multiplicative congruential generator $k \leq m-1$ – since if 0 occurs it repeats indefinitely.
- If $k = m$ then we say that the generator has full period. If a multiplicative generator has period $m-1$, this is called maximal.

Note

Having full or maximal period does not, on its own, ensure a good generator. For example set $a = b = 1$ – giving the sequence $0, 1, 2, 3, \ldots, m-1$ (highly predictable, does not resemble randomness!).

1.3 Choice of a, b, m and X_0

We want...

(A) ...to make the arithmetic easy

"By hand", $m = 10^\beta$ is easy, for example

$$2794321 \bmod(10^5) \equiv 94321.$$

"On computer", $m = 2^\beta$ (Binary) is better: e.g.

$$\underbrace{1110101}_{\text{discard}}11010101 \bmod(2^8) \equiv 11010101.$$

(B) ...to have long cycle-length

The period k depends on X_0, a, b and m. For example,

$$X_{n+1} \equiv (5X_n + 4)\bmod(2^4) \quad \text{i.e. } a = 5, b = 4, m = 2^4 = 16$$

	X_1	X_2	X_3	X_4	X_5	X_6	
$X_0 = 0$	4	8	12	0	4	8	$k = 4$
$X_0 = 1$	9	1	9	1	9	1	$k = 2$
$X_0 = 3$	3	3	3	3	3	3	$k = 1$

CG: $X_{n+1} = (aX_n + b) \bmod m$

We quote some theorems concerning the choice of a, b, X_0 and m; the proofs are not examinable but, for those interested, can be found in Ripley (1987).

Theorem 1 *A mixed congruential generator has full period m iff:*

(i) $\gcd(b, m) = 1$.

(ii) $a \equiv 1 \bmod p$ *for each prime factor* p *of* m.

(iii) $a \equiv 1 \bmod 4$ *if* 4 *divides* m.

Corollary 1 *If m is prime, we obtain full period only if $a = 1$.*

Theorem 2 *A multiplicative congruential generator with $m = 2^\beta (\geq 16)$ has largest period $m/4$. This is attained iff*

$$a \bmod 8 \equiv 3 \text{ or } 5, \quad \text{and } X_0 \text{ is odd}$$

Theorem 3 *A multiplicative congruential generator has period $m - 1$ only if m is prime.*

Notes

1. From Theorem 1 (i.e. if $b \neq 0$, for mixed generators), if $m = 2^\beta$ $(\beta \geq 2)$ then prime factor is 2 and 4 divides m, so (iii) implies $a = 4c + 1$ for some c. So (ii) is automatically satisfied. To satisfy (i) take b odd.

2. From Theorem 2 (i.e. for multiplicative generators), note that we can choose $a = 5^{2q+1}$ from $q \in \{0, 1, 2, \ldots\}$ since

$$\begin{aligned}
5^{2q+1} &= (1+4)^{2q+1} = \textstyle\sum_{i=0}^{2q+1} \binom{2q+1}{i} 4^i 1^{2q+1-i} \text{ (Binomial theorem)} \\
&= 1 + 4(2q+1) + \tfrac{4^2(2q+1)2q}{2} + \ldots + 4^{2q+1} \\
\Rightarrow a \bmod 8 &= (1 + 4(2q+1)) \bmod 8 \\
&= (1 + 8q + 4) \bmod 8 \\
&= 5 \bmod 8 \quad \text{as required}
\end{aligned}$$

So multiplicative generators of the form $X_{n+1} \equiv 5^{2q+1} X_n \text{mod}(2^\beta)$ attain period $2^{\beta-2}$ if X_0 is odd. (Note: this works for $a = 13^{2q+1}$ also). So, we can ensure large periods with multiplicative generators. The NAG subroutine `G05CAF` uses a multiplicative generator with $a = 13^{13}, m = 2^{59}$, giving a period of $2^{57} \approx 6 \times 10^{17}$.

Example

Does the congruential generator with the following values have full period:

$$X_0 = 5772156648, \qquad a = 3141592621,$$
$$b = 2718281829, \qquad m = 10^{10}?$$

We use Theorem 1.

- Prime factors of m are $2, 5 \Rightarrow \gcd(b, m) = 1$ (i) ✓
- $a \equiv 1 \bmod 2$, $a \equiv 1 \bmod 5 \Rightarrow$ (ii) ✓
- $a \equiv 1 \bmod 4$, 4 divides $m \Rightarrow$ (iii) ✓

The above theorems show us how to find generators with a long period, but can we take seriously a sequence with finite period? Consider the multiplicative congruential generator $X_{n+1} \equiv 5^{17} X_n \bmod (2^{43})$, which has a period of $2^{41} \approx 2 \times 10^{12}$. If we were to generate 1000 numbers per second, the sequence would not repeat for more than 63 years.

We have already demonstrated (by setting $a = b = 1$ / $a = b = 0$) that a large period is not enough to ensure a ~~large~~ good generator; we must also ensure that the statistical properties of the generated sequence mimic that of independent Uniforms.

Notes

1. Regardless of how large the period is, we will find numbers repeating when we truncate to a limited number of decimal places. This is often governed by machine precision.

2. Often the seed is chosen by the computer via internal clock.

3. We will also consider the Wichmann-Hill generator (Whichmann & Hill, Applied Statistics 1982) - the implementation of which is the subject of Exercises 3, Q1. This generator has a cycle length $> 2.78 \times 10^{13}$, so if we used 1000/second it would take more than 880 years for the sequence to repeat.

2 Testing Random Numbers

We have seen how we can obtain sequences with large period, but do such sequences produce numbers that "look like" realizations of an independent uniform random variable?

We will concentrate on empirical tests (*i.e.* tests of samples from the generator), rather than theoretical tests.

There are different types of test that we can use

- Diagnostic plots (e.g histogram)
- Examine sample moments of the output.
- test how often certain digits occur
- test dependence between consecutive pairs, triplets, etc... of variates.

Notes

1. By the nature of random digits, certain sequences will certainly fail tests of uniformity.

2. Over a full cycle, properties may have been shown to be good, but this may not necessarily be true over particular sequences.

3. Theoretical tests possible for testing dependence of successive k-tuples for small k (Ripley, 1987).

4. Tests are also useful as a check that the generator has been programmed/implemented correctly.

5. Test can be used for specific distributions, not just Uniforms.

2.1 Check of moments

Generate a sequence of size n from stated distribution. Consider an independent sample $X_1, X_2, \ldots, X_n$ from pdf $f_X(x)$, with mean μ and variance σ^2. Then the CLT says that

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

has an asymptotic $\mathrm{N}\left(\mu, \frac{\sigma^2}{n}\right)$ distribution. We could then use large sample techniques, for example in the case of the Uniform distribution:

$$H_0 : \mu = \tfrac{1}{2}$$
$$H_0 : \sigma^2 = \tfrac{1}{12}$$

Aside: for a full period congruential generator, over the whole period, we have the sample:

$$x_i = \frac{i}{m} \quad i = 0, 1, 2, \ldots, m-1.$$

Sample mean:

$$\bar{x} = \frac{1}{m} \sum_{i=0}^{m-1} \frac{i}{m}$$

$$= \frac{1}{2}\left(1 - \frac{1}{m}\right)$$

Sample variance:

$$s^2 = \frac{1}{m-1} \sum_{i=0}^{m-1} \left(\frac{i}{m} - \bar{x}\right)^2$$

$$= \frac{1}{12}\left(1 + \frac{1}{m}\right).$$

ie the larger the period, the closer the sample moments are to the theoretical moments for $U(0,1)$.

2.2 Time series methods

Let $X_1, X_2, \ldots X_n$ be a sequence of random variables. The sample autocorrelation sequence is given by

$$\hat{\rho}_k = \frac{\frac{1}{n}\sum_{t=1}^{n-k}(X_t - \bar{X})(X_{t+k} - \bar{X})}{\frac{1}{n}\sum_{t=1}^{n}(X_t - \bar{X})^2} :$$

plotting the sample autocorrelation against k gives the *correlogram.* It can be shown (Kendall & Stuart, 1969 (Advanced Theory of Statistics)) that if $X_1, X_2, \ldots, X_n$ are IID random variables with arbitrary mean,

$$\mathbb{E}(\hat{\rho}_k) \approx -\frac{1}{n} \qquad \text{and} \qquad \mathrm{var}(\hat{\rho}_k) \approx \frac{1}{n}.$$

So an approximate 95% confidence interval for the estimated autocorrelation sequence is

$$-\frac{1}{n} \pm \frac{1.96}{\sqrt{n}};$$

this interval can be drawn on the correlogram. The R function `acf()` produces the correlogram, with confidence intervals.

2.3 Diagnostic plots

Various plots can aid the detection of problems with the generated sequence, e.g.:

- U_i against U_{i+k} with $k = 1, 2, \ldots$ should be a random scatter;
- A histogram of the sequence should approximate the theoretical density function.

2.4 Tests for random digits

Tests can be applied to the U_i, or alternatively to $X_i = \lfloor KU_i \rfloor$, suitable K. Taking $K = 10$ gives X_i digits in $0 - 9$.

Recall: elementary chi-squared goodness-of-fit test.

Suppose we have a sequence $X_1, X_2, \ldots, X_n$, with outcomes falling into N possible categories. Denote the observed frequencies in each category $O_1, O_2, O_3, \ldots, O_N$, noting $\sum O_i = n$. Denote the expected frequencies under some H_0 for the distribuion of variates by $E_1, E_2, E_3 \ldots, E_N$, with $\sum E_i = n$.

To measure departure of the observed from the expected, use the following statistic

$$S = \sum_{i=1}^{N} \frac{(O_i - E_i)^2}{E_i}$$

Intuitively, if this statistic is BIG - suspect H_0. If n is large, then S typically follows a chi-squared distribution with $N - 1$ degrees of freedom, i.e.

$$\sum_i \frac{(O_i - E_i)^2}{E_i} \sim \chi^2_{N-1}.$$

Thus we compare the observed statistic to the chi-squared distribution as a method of testing the hypothesis.

2.4.1 Frequency test of digits

Given a sequence of length n, representing, for example, the first decimal place of our generated uniforms, we can examine the number of times each of the digits $0-9$ occur in the sequence. In this case we have

$$O_0, O_1, \ldots, O_9$$
$$E_0 = \tfrac{n}{10}, E_1 = \tfrac{n}{10}, \ldots, E_9 = \tfrac{n}{10}$$

Then we can compare the statistic

$$S = \sum_{i=0}^{9} \frac{(O_i - \frac{n}{10})^2}{\frac{n}{10}}$$

with the tails of χ^2_9.

This gives no information about the possible dependence structure.

2.4.2 Serial test

Let n_{jk} be the number of times the digit j is followed by the digit k. In this case we have

$$O_0 = n_{00}, O_1 = n_{01}, \ldots, O_9 = n_{09}, O_{10} = n_{10}, \ldots, O_{99} = n_{99}$$
$$E_0 = \tfrac{n}{100}, E_1 = \tfrac{n}{100}, \ldots, E_{99} = \tfrac{n}{100}$$

Then we can compare the statistic

$$\sum_{j=0}^{9}\sum_{k=0}^{9} \frac{(n_{jk} - \frac{n}{100})^2}{\frac{n}{100}}$$

with the tails of χ^2_{99}.

BUT

(a) Hard work.

(b) Higher order dependence.

2.4.3 Gap test

Choose a digit, say 3, then record the length of the subsequence lying between occurrences of the digit 3.

$$X_1, X_2, X_3, 3, \underbrace{X_5, X_6, \ldots, X_{k+4}}_{\text{gap length= k}}, 3, \ldots$$

If the sequence is uniform and independent, the distribution of gap lengths, K, should be Geometric $(\frac{1}{10})$

Gap length (k)	0	1	2	3	...
$P(K=k)=p_k$	$\frac{1}{10}$	$\frac{9}{10}\frac{1}{10}$	$(\frac{9}{10})^2\frac{1}{10}$	$(\frac{9}{10})^3\frac{1}{10}$	...
	p_0	p_1	p_2	p_3	

For example,

Gap length	0	1	2	3	4	5	>5
Observed freq.	O_0	O_1	O_2	O_3	O_4	O_5	$O_{>5}$
Expected freq.	np_0	np_1	np_2	np_3	np_4	np_5	$np_{>5}$

where $n = \sum_i O_i$; $p_{>5} = 1 - p_0 - p_1 - p_2 - p_3 - p_4 - p_5$

Then we can compare the statistic

$$S = \sum_i \frac{(O_i - np_i)^2}{np_i}$$

with the tails of χ^2_6. ← 6

2.5 Conclusions

1. Better to use a well-documented generator (see Ripley (1987) for such generators).

2. Preferable to have at least two alternative generators to repeat simulation study with.

3. Any tests will be failed occasionally by chance. In large investigations it is a good idea to try each test on a large number of non-overlapping subsequences.

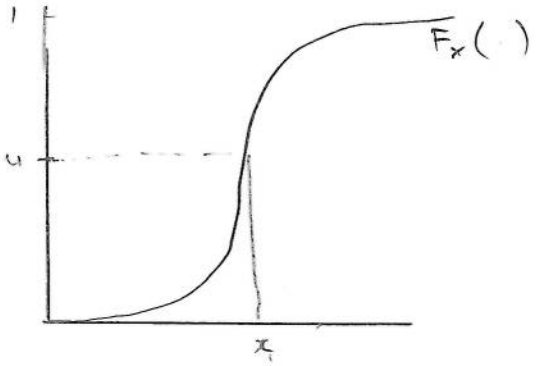
4. While good generators may occasionally fail some of the tests, it is also true that a poor generator may pass them.

3 General Methods for Random Variate Generation

We now assume that we have a stream of independent uniforms and wish to generate from more general distributions.

3.1 Inversion

If $X \sim F_X$ (continuous) $\Rightarrow U = F_X(X) \sim \mathrm{U}(0,1)$.



Proof (sketch)

$$\begin{aligned} F_U(u) &= \mathrm{P}(U \le u) = \mathrm{P}(F_X(X) \le u) \\ &= \mathrm{P}(X \le F_X^{-1}(u)) = F_X(F_X^{-1}(u)) = u \quad \text{c.d.f. of a uniform.} \end{aligned}$$

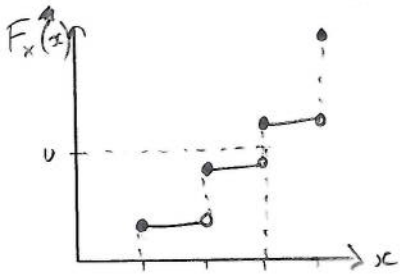
This result is known as the probability integral transform, and allows us to transform any continuous random variable into a uniform random variable...and vice versa! We therefore have the following method of generating $X_i \sim F_X$:

1. generate $U_i \sim \mathrm{U}(0,1)$;
2. set $X_i := F_X^{-1}(U_i)$.

Note that a key requirement above is the existence of $F_X^{-1}(\cdot)$.

Consider discrete dist?

x	1	2	3	4
$P(X=x)$	0.1	0.3	0.2	0.4



For arbitrary RVs (ie not necessarily continuous), we can use the generalized inverse distribution function

$$F_X^-(u) = \min\{x : F_X(x) \geq u\}$$

and replace step 2 above with "Set $X_i = F_X^-(U_i)$"

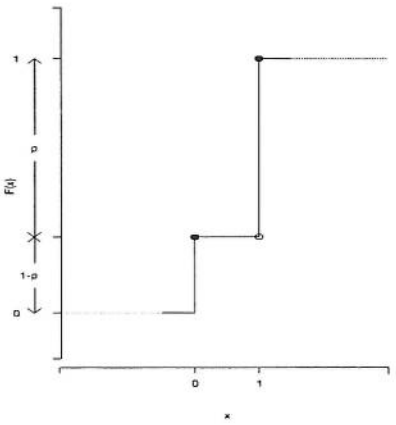
Note also that, for an independent sample $X_1, \ldots, X_n$, we simply start with an independent sample $U_1, \ldots, U_n$.

Examples

a. Want $X \sim \text{Bernoulli}(p)$, *i.e.*,

$$X = \begin{cases} 1 & \text{w.p.} \quad p \\ 0 & \text{w.p.} \quad 1-p \end{cases}$$

i.e. $\mathrm{P}(X = 1) = p$ and $\mathrm{P}(X = 0) = 1 - p$.



Algorithm (Bernoulli)

1. Generate $U = u \sim U(0, 1)$.
2. If $u > 1 - p$, set $X = 1$,
   else set $X = 0$.
   Then $X \sim \text{Bernoulli}(p)$.

b. Want $X \sim \exp(\lambda)$. So,

$$F_X(x) = 1 - e^{-\lambda x}.$$

Set $U = 1 - e^{-\lambda X} \Rightarrow X = -\frac{1}{\lambda}\log(1-U)$

$\Rightarrow F_X^{-1}(U) = -\lambda^{-1}\log(1-U)$.

Algorithm (Exponential)

1. Generate $U = u \sim U(0,1) \Rightarrow 1 - U \sim U(0,1)$.
2. Set $X = -\lambda^{-1}\log(u)$. Then $X \sim \exp(\lambda)$.

c. Want $X \sim$ Cauchy, i.e.

$$f_X(x) = \frac{1}{\pi(1+x^2)}$$

$f_X(x;\mu,\sigma^2) = \frac{1}{\pi\left(1+\left(\frac{x-\mu}{\sigma}\right)^2\right)}$

μ - median mode.

$\mathbb{E}[X] = \infty$

$V[X] = \infty$

$F_X(x) = \int_{-\infty}^{x} \frac{1}{\pi(1+y^2)}\,dy = \frac{1}{2} + \frac{1}{\pi}\tan^{-1}(x)$

$\Rightarrow F_X^{-1}(U) = \tan\left[\pi\left(U - \frac{1}{2}\right)\right]$

Algorithm (Cauchy)

1. Generate $U \sim U(0,1)$.
2. Set $X = \tan\left[\pi\left(U - \frac{1}{2}\right)\right]$

d. Imagine that we want to generate $X \sim N(0,1)$. Note that

$$F_X(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}y^2}\,dy$$

But $F_X^{-1} = ?$. This can only be solved numerically.

This is a severe limitation on the inversion method.

3.2 Rejection

Suppose we wish to generate X with a known pdf $f_X(\cdot)$, but that this is difficult (i.e. we cannot use inversion).

If, ~~however~~, it is possible to generate a different RV Y from a distribution with known pdf g_Y, this may be enough, as long as:

- the support of g_Y encompasses that f_X ie $f_X(x) > 0 \Rightarrow g_Y(x) > 0$
- there exists $M > 0$ such that $\forall x, f_X(x) > 0$

$$\frac{f_X(x)}{g_Y(x)} \leq M < \infty$$

~~we are able to sample from a However, if it is easy to sample from Y with pdf g, and f, g are s.t. $\exists M > 0$, with $\frac{f}{g} \leq M < \infty$.~~

Algorithm Outline (rationale later):

1. Generate Y from g_Y
2. For some function $h(\cdot)$ with values in [0,1], given $Y = y$, set $X = y$ with probability $h(y)$, otherwise return to 1. ie we reject y with prob $1-h(y)$.

3.2.1 Properties of this procedure

What is the distribution of the accepted RV X? We seek $\mathbb{P}[Y \leq x \mid X \text{ is set} = Y]$

$$\mathbb{P}[(Y \leq x) \cap (X \text{ is set} = Y)] = \int_{-\infty}^{x} h(y)\, g_Y(y)\, dy$$

$$\Rightarrow \mathbb{P}[X \text{ is set} = Y] = \int_{-\infty}^{\infty} h(y) g_Y(y) dy$$

$$\Rightarrow \mathbb{P}[(Y \leq x) \mid X \text{ is set} = Y] = \frac{\int_{-\infty}^{x} h(y) g_Y(y) dy}{\int_{-\infty}^{\infty} h(y) g_Y(y) dy}$$

$$\Rightarrow \text{pdf of accepted } X\text{s} = \frac{h(x) g_Y(x)}{\int_{-\infty}^{\infty} h(y) g_Y(y) dy}$$

What if we choose

$$h(y) = \frac{f(y)}{M g(y)}?$$

$$\Rightarrow \text{pdf of accepted } X\text{s} \quad \frac{(f/Mg)\, g}{\int \frac{f}{Mg}\, g} = \frac{f_X(x)}{\int f_X(y) dy} = f_X(x)$$

Rejection Sampling Algorithm

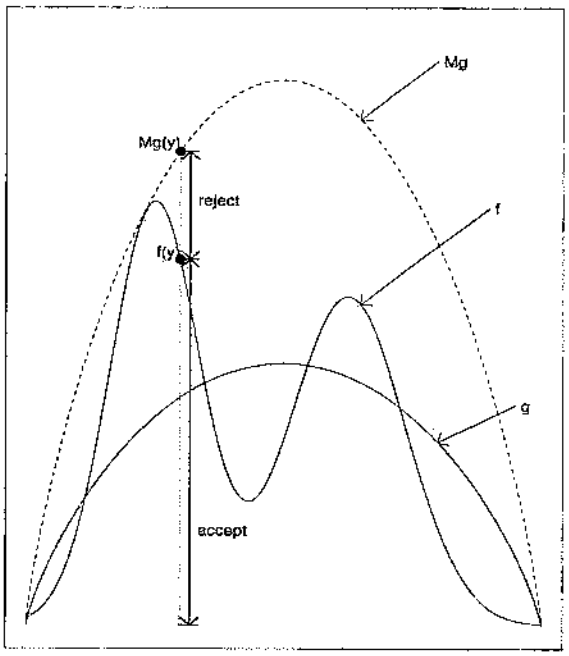
1. Generate $Y = y \sim g(.)$.
2. Generate $U = u \sim U(0,1)$.
3. If $u \leq \frac{f(y)}{Mg(y)}$ set $X = y$.
4. Otherwise GOTO 1.

$$\frac{f_X(y)}{M g_Y(y)} \in [0, 1]$$

Here

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

Mg is an "envelope" for f. The condition $u \leq \frac{f(y)}{Mg(y)}$ implies that $Mg(y)u \leq f(y)$, now $Mg(y)u$ is a point "at random" below $Mg(y)$, we accept this point if it lies below $f(y)$.



3.2.2 How many "goes" to accept a value?

Let Z = number of rejections before accepting X.

Let $p = \mathrm{P}(\text{accept } X \text{ at each attempt})$,

$\mathrm{P}(Z = k) = p(1-p)^{k-1}$, $k = 0, 1, 2, \ldots$, *i.e.* $Z \sim \text{Geometric}(p)$.

$$\Rightarrow \mathrm{E}[\text{number of rejections per accepted } X \text{ variate}] = \mathrm{E}[Z] = \frac{1-p}{p}.$$

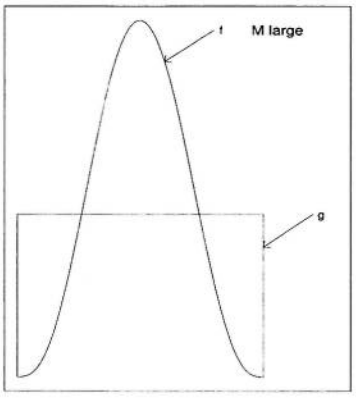
$$p = \int_{-\infty}^{\infty} h(y)g(y)\,dy.$$

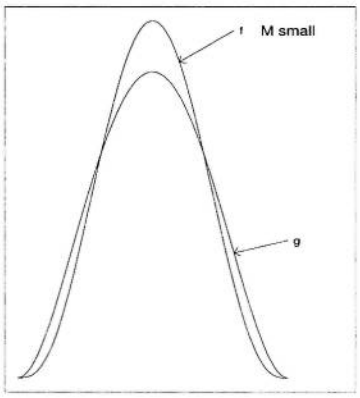
$$= \int_{-\infty}^{\infty} \frac{f(y)}{M g(y)} g(y)\,dy$$

$$= \frac{1}{M}\int_{-\infty}^{\infty} f(y)\,dy = \frac{1}{M}$$

$$\Rightarrow E[Z] = M - 1$$

So, ideally, we want to have M small. This can be achieved through prudent choice of g: the more the shape of g mimics the shape of f, the more efficient the procedure will be.





$\ominus$ Can almost always find a suitable g, except perhaps if f is unbounded or if tails of f are heavy – in both cases, we can't always find $g(\cdot)$ such that $\frac{f(x)}{g(x)} < M \forall x$.

⊖ In order for the procedure to be efficient:

(i) we need to understand the shape of f_X in detail, to choose g_Y to mimic f_X;

(ii) it should be easy to simulate from g_Y.

Typically, sampling distributions $g_Y(\cdot)$ for which exact sampling is straightforward require large M. — TRADEOFF

3.2.3 Extension

Suppose our target pdf $f_X(\cdot)$ can be written

$$f_X(x) = \frac{f_X^*(x)}{\int f_X^*(y)dy}.$$

If we choose $h(y) = \frac{f^*(y)}{Mg(y)}$, we can proceed almost exactly as before, noting only that

$$\text{P[accept } X] = \int_{-\infty}^{\infty} \frac{f^*(y)}{Mg(y)} g(y)\,dy = \frac{\int_{-\infty}^{\infty} f^*(y)\,dy}{M}$$

so we should instead test $u \leq \frac{f^*(y)}{Mg(y)}$ where $y \sim g(\cdot)$ and $M = \sup_x \frac{f^*(x)}{g(x)}$.

Hence, we only need to know f "up to proportionality".

Also only need to know the form of g up to proportionality: if $g \propto g^*$

$M = \sup_x \frac{f^*(x)}{g^*(x)}$ and test $u \leq \frac{f^*(y)}{Mg^*(y)}$.

Examples of working with f*

(i) Generate from a normal using Cauchy as rejection envelope function.

$$f^*(x) = e^{-x^2/2}, \quad g^*(x) = (1+x^2)^{-1}.$$

Want, $\sup_x \frac{f^*(x)}{g^*(x)}$.

Let $y = \log\left(\frac{f^*(x)}{g^*(x)}\right) = -\frac{x^2}{2} + \log\left(1+x^2\right)$.

$$\frac{dy}{dx} = x\left(\frac{2}{1+x^2} * -1\right) = 0 \Rightarrow x = 0 \text{ or } x = \pm 1$$

$$\frac{d^2y}{dx^2} = -1 + \frac{2-2x^2}{(1+x^2)^2} \Rightarrow \begin{cases} x=0 \Rightarrow y'' > 0 \Rightarrow \min \\ x=\pm 1 \Rightarrow y'' < 0 \Rightarrow \max \end{cases}$$

So set

$$M = \sup_x \frac{f^*(x)}{g^*(x)} = \frac{f^*(1)}{g^*(1)} = 2e^{-\frac{1}{2}}$$

Algorithm

1. Generate $U_1 = u_1 \sim U(0,1)$.
2. Set $y = \tan[\pi(u_1 - \frac{1}{2})] \Rightarrow y$ is Cauchy.
3. Generate $U_2 = u_2 \sim U(0,1)$.
4. Check if $u_2 \leq \frac{e^{-y^2/2}(1+y^2)}{2e^{-\frac{1}{2}}}$; if it is, set $X = y \Rightarrow X \sim N(0,1)$.
5. Otherwise Goto 1.

Acceptance Probability

$$\begin{aligned} \mathrm{P}(\text{accept } X) &= \int_{-\infty}^{\infty} \frac{f^*(y)}{Mg^*(y)} g(y)\,\mathrm{d}y, \\ &= \frac{1}{M} \int_{-\infty}^{\infty} \frac{e^{-y^2/2}}{(1+y^2)^{-1}} \left(\pi(1+y^2)\right)^{-1} dy. \\ &= \frac{e^{1/2}}{2\pi} \int_{-\infty}^{\infty} e^{-y^2/2}\, dy = \frac{e^{1/2}}{\sqrt{2\pi}} \approx 0.657. \end{aligned}$$

(ii) Want $X \sim \mathrm{Beta}(\alpha, \beta)$.

$$\begin{aligned} f(x) &= \begin{cases} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}, & 0 < x < 1 \\ 0, & \text{o/w} \end{cases} \\ &\propto \underbrace{x^{\alpha-1}(1-x)^{\beta-1}}_{f^*} \end{aligned}$$

Clearly easier to work with f^* than f.

Example of Implementation

(a) Work with f^*.

(b) "Lazy" choice of g? $U(0,1)$.

(c) Identify $M = \sup_x \frac{f^*}{g}$.

$$\frac{f^*(x)}{g(x)} = x^{\alpha-1}(1-x)^{\beta-1}$$

Setting $\frac{\mathrm{d}}{\mathrm{d}x} = 0$, gives maximum when $x = \frac{\alpha-1}{\alpha+\beta-2}$

$$\Rightarrow M = \frac{(\alpha-1)^{\alpha-1}(\beta-1)^{\beta-1}}{(\alpha+\beta-2)^{\alpha+\beta-2}}$$

(iii) Sampling from Bayesian posterior densities.

$$\begin{aligned} f(\theta|D) &= \frac{p(\theta)f(D|\theta)}{\int p(\theta)f(D|\theta)} \\ &\propto p(\theta)f(D|\theta). \end{aligned}$$

← common not to have the normalising constant.

e.g. Let θ = parameter $\quad p(\theta)$ = 'prior distribution'

$$f(D|\theta) = l(\theta) = \text{'likelihood'}.$$

Then the 'posterior' $f(\theta|D) \equiv f(\theta) \propto l(\theta)p(\theta)$, *i.e.* $f^*(\theta) = l(\theta)p(\theta)$.

Suppose we use prior as our distribution to simulate from (i.e. $g \equiv p$).

We require

$$M = \sup_\theta \frac{f^*(\theta)}{p(\theta)} = \sup_\theta l(\theta) = l(\hat{\theta}),$$

$\hat{\theta}$ = MLE.

ie M is equal to the maximised likelihood

Algorithm

1. Generate $U = u \sim U(0,1)$ and $\theta \sim p(x)$.
2. If

$$u \leq \frac{f^*(\theta)}{Mg(\theta)} = \frac{l(\theta)}{l(\hat{\theta})}, \quad \text{accept } \theta \Rightarrow \theta \sim f(\cdot).$$

3. Otherwise GOTO 1.

3.3 Box-Müller method for generating N(0,1)'s

Suppose we have access to two independent, uniform variates:

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

We consider two successive transformations:

$$\begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \longmapsto \begin{pmatrix} R := (-2\log U_1)^{1/2} \\ A := 2\pi U_2 \end{pmatrix} \longrightarrow \begin{pmatrix} X := R\cos A, \\ Y := R\sin A. \end{pmatrix}$$

Box and Müller (1958) showed that X and Y are independent N(0,1)'s.

Note that A and R are independent also.

Algorithm

1. Generate $U_1 \sim U(0,1)$, $U_2 \sim U(0,1)$.
2. Set $R = (-2\log U_2)^{1/2}$, $A = 2\pi U_1$.
3. Set $X = R\cos A$, $Y = R\sin A$, then $X, Y \sim \mathrm{N}(0,1)$.

Problem: cos and sin are "expensive" to evaluate.

3.4 Marsaglia's polar method for generating N(0,1)'s

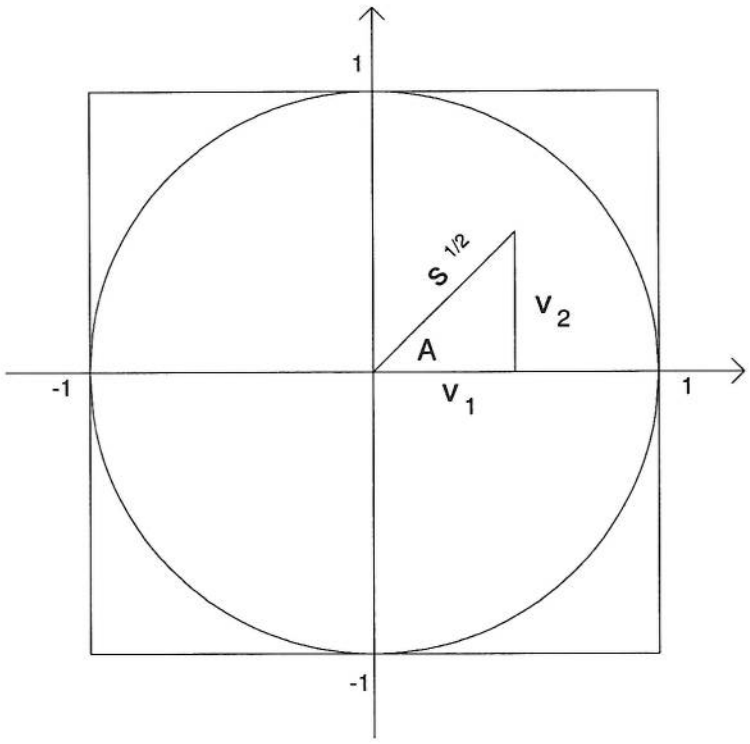
Marsaglia and Bray (1964) proposed the following ingenious alternative.

Generate U_1, U_2 independent $U(0,1)$'s, and set $V_i = 2U_i - 1, i = 1,2$. This gives random points in the square $[-1,1] \times [-1,1]$. Now consider points in the unit circle only, i.e. reject those points with $V_1^2 + V_2^2 > 1$.

Now consider (S, A) where

$$\begin{aligned} S &= V_1^2 + V_2^2 \\ A &= \tan^{-1}\left(\frac{V_2}{V_1}\right). \end{aligned}$$

These are simply the polar coordinates of the random point in the unit circle, and we note that S and A are independent with $S \sim U(0,1)$ and $A \sim U(0, 2\pi)$.



Recall for Box-Müller we set $X = R\cos A$ and $Y = R\sin A$, where R is a transformed Uniform variate and $A \sim U(0, 2\pi)$. We can therefore obtaining the required R straightforwardly:

$$R := (-2\log S)^{1/2}.$$

More importantly, we can use standard trigonometric relationships to obtain $\cos A$ and $\sin A$ in terms of $\tan A$, and therefore in terms of V_1 and V_2:

$$\cos A = \frac{1}{\sqrt{\tan^2 A + 1}} \Rightarrow \cos A = \frac{V_1}{\sqrt{V_1^2 + V_2^2}} \; ; \; \sin A = \frac{V_2}{\sqrt{V_1^2 + V_2^2}}$$

So we have

$$X = (-2\log S)^{1/2}\cos A \quad Y = (-2\log S)^{1/2}\sin A$$

$$X = \sqrt{\frac{-2\log S}{S}}\, V_1 \; ; \; Y = \sqrt{\frac{-2\log S}{S}}\, V_2$$

resulting in two independent $\mathrm{N}(0, 1)$ variates.

Algorithm

1. Generate $U_i = u_i \sim U(0,1)$ for $i = 1, 2$.

2. Set $V_i = 2u_i - 1$

3. If $S = V_1^2 + V_2^2 \leq 1$

   Let

$$C = \sqrt{-\frac{2}{S}\log S}, \quad \text{set} \quad \begin{aligned} X &= CV_1, \\ Y &= CV_2. \end{aligned}$$

   Then $X, Y \sim$ N(0,1), independently.

4. Otherwise GOTO 1

Note that the rejection step is not too inefficient since the probability of accepting a (V_1, V_2) point is $\frac{\pi}{4} \approx 0.785$.

3.5 Composition

Consider

$$f = \pi_1 f_1 + \ldots + \pi_k f_k$$

where $\pi_i \geq 0, \sum \pi_i = 1$, and where f_i is a valid density, $i = 1, \cdots, k$.

To generate a r.v. from $f = \pi_1 f_1 + \ldots + \pi_k f_k$:

- Pick i with probability π_i.

- Then generate from f_i (e.g. using rejection).

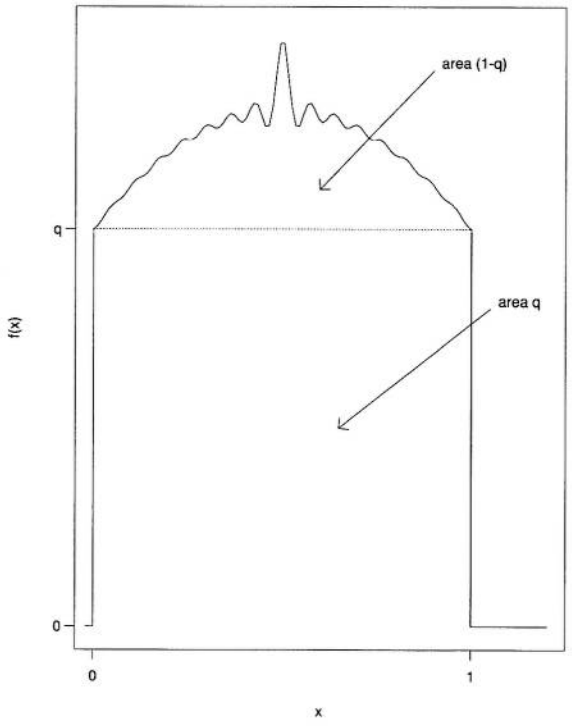
Why?

PDF might actually be of this form

- may be some composition of 2 or more densities at each point in a common domain.
- pdf may behave differently over different parts of the domain.

Example

(i) Consider the following density, $f(x)$:



← Horizontal split'.

To simulate from $f(x)$, we can write,

proper density for the irregular part.

$$f(x) = q + \{f(x) - q\}$$

$$= q \times 1 + (1-q)\left\{\frac{f(x)-q}{1-q}\right\}$$

$U(0,1)$ density

Approach

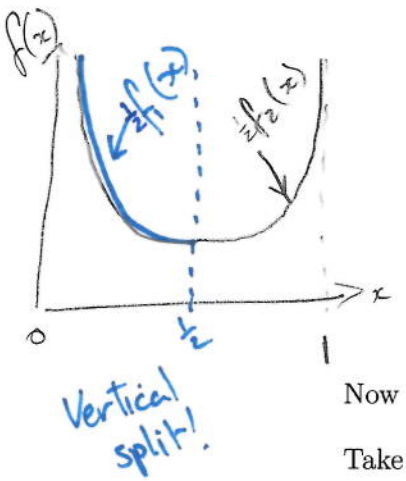
- With prob q, sample from $U(0,1)$.
- With prob $(1-q)$ simulate from

$$f_1(x) = \frac{f(x)-q}{1-q}$$

NB: If $(1-q)$ is small we are less likely to need to sample from the more complicated $f_1(x)$.

(ii) Consider $X \sim \text{Beta}(\alpha, \beta), \quad \alpha = \beta < 1.$

Think of $f(x) = \frac{\Gamma(2\alpha)}{\Gamma(\alpha)\Gamma(\alpha)} x^{\alpha-1}(1-x)^{\alpha-1}$ as $\frac{1}{2}f_1(x) + \frac{1}{2}f_2(x)$, where



Vertical split!

$$f_1(x) = \begin{cases} \frac{2\Gamma(2\alpha)}{\Gamma(\alpha)\Gamma(\alpha)} x^{\alpha-1}(1-x)^{\alpha-1} & \text{on } \left(0, \frac{1}{2}\right] \\ 0 & \text{on } \left(\frac{1}{2}, 1\right) \end{cases}$$

$$f_2(x) = \begin{cases} 0 & \text{on } \left(0, \frac{1}{2}\right] \\ \frac{2\Gamma(2\alpha)}{\Gamma(\alpha)\Gamma(\alpha)} x^{\alpha-1}(1-x)^{\alpha-1} & \text{on } \left(\frac{1}{2}, 1\right) \end{cases}$$

NB: both f_1 & f_2 are proper.

Now think of suitable g's for rejection sampling for f_1 and f_2.

Take $g_1(x) \propto x^{\alpha-1}$

$$g_1(x) = \begin{cases} kx^{\alpha-1} & x \in \left(0, \frac{1}{2}\right] \\ 0 & x \in \left(\frac{1}{2}, 1\right) \end{cases}$$

$g_2(x)$ similar – take $g_2(x) \propto (1-x)^{\alpha-1}$

Generation from g_1?

INVERSION:

$$G_1(x) = \int_0^x k t^{\alpha-1} dt = \begin{cases} 0 & x \le 0 \\ \frac{k}{\alpha} x^{\alpha} & x \in \left(0, \frac{1}{2}\right] \\ \frac{k}{\alpha 2^{\alpha}} & x > \frac{1}{2} \end{cases}$$

So take $k = \alpha 2^{\alpha}$

$$\Rightarrow G^{-1}(U) = \frac{1}{2} U^{\frac{1}{\alpha}}$$

~~Algorithm~~ Approach.

1. With probabilities $\frac{1}{2}, \frac{1}{2}$ pick f_1 or f_2.
2. for e.g. f_1 use rejection technique based on g_1, where we generate from g_1 using inversion.

3.6 Ratio of Uniforms

Suppose we have some distribution, with density known up to proportionality, i.e. we have a function h with $h(\cdot) \geq 0$ and $\int h < \infty$, and we are interested in sampling from $h \Big/ \int h$.

Consider the region in (U, V) space defined by:

$$C_h = \left\{ (u, v) \mid 0 \leq u \leq \sqrt{h\left(\frac{v}{u}\right)} \right\}$$

We can show that C_h has finite area, and that if (U, V) are uniform on C_h, then $X = \frac{V}{U}$ has density $\frac{h}{\int h}$.

Proof

$$\text{Area}(C_h) = \iint_{C_h} du\,dv \qquad \left((U,V) \to \left(U, X = \frac{V}{U}\right)\right).$$

$$= \int_0^{\sqrt{h(x)}} \int u\,du\,dx \qquad V = XU \Rightarrow \frac{dv}{dx} = u$$

$$= \frac{1}{2}\int h(x)\,dx < \infty \qquad 0 \leq u \leq \sqrt{h(x)}$$

Seek $f_X(x)$

$$f_{U,V}(u,v) = \begin{cases} \frac{1}{\text{Area}(C_h)} & (u,v) \in C_h \\ 0 & \text{otherwise} \end{cases}$$

$$f_{U,V}(u,v) = f_{U,X}(u,x) \left| \frac{\partial(u,x)}{\partial(u,v)} \right| \Rightarrow u f_{U,V}(u,v) = f_{U,X}(u,x)$$

$$f_X(x) = \int_u f_{U,X}(u,x)\,du = \int_0^{\sqrt{h(x)}} \frac{u}{\text{Area}(C_h)}\,du$$

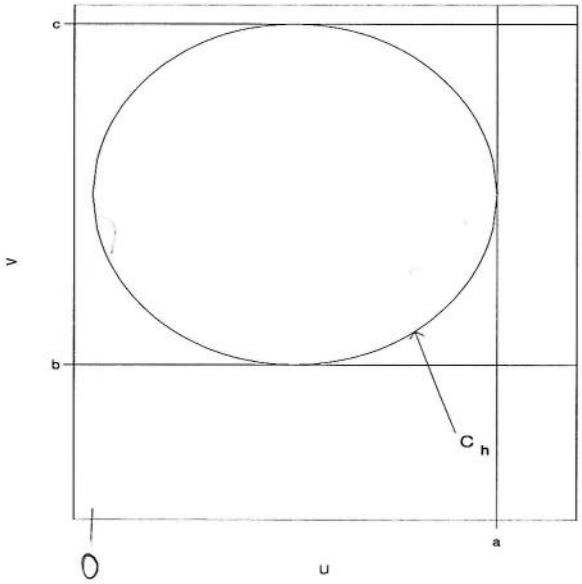
$$= \frac{\frac{1}{2}h(x)}{\frac{1}{2}\int h(x)\,dx}$$

NB: can use an exact density if we like, but we only *need* it up to proportionality.

We need to generate numbers within C_h: one way of doing this is to bound the C_h region within a rectangle.

3.6.1 Bounding rectangle

$$C_h = \left\{ (u, v) \mid 0 \leq u \leq \sqrt{h\left(\frac{v}{u}\right)} \right\}$$



We find the dimensions of the bounding rectangle

To find a: Write $h(x)$.

$$0 \leq u \leq \sup_x \sqrt{h(x)} := a$$

To find b, c:

$$x = \frac{v}{u} \Rightarrow u = \frac{v}{x} \leq \sqrt{h(x)}$$

$$x \leq 0 \Rightarrow v \geq x\sqrt{h(x)} \Rightarrow v \geq \inf_{x \leq 0} x\sqrt{h(x)} := b$$

$$x \geq 0 \Rightarrow v \leq x\sqrt{h(x)} \Rightarrow v \leq \sup_{x \geq 0} x\sqrt{h(x)} := c$$

So, set

$$\boxed{a = \sup_x \sqrt{h(x)} \quad b = \inf_{x \leq 0} x\sqrt{h(x)} \quad c = \sup_{x \geq 0} x\sqrt{h(x)}}$$

Such a rectangle will always exist provided $h(x)$ and $x^2 h(x)$ are bounded in the domain of x.

Algorithm

1. Find bounding rectangle for C_h (*i.e* find a, b and c).
2. Generate $(U_1, U_2) \sim U(0,1)$.
3. Set $U = aU_1$, $V = b + (c-b)U_2$.
4. if $U \leq \sqrt{h\left(\frac{V}{U}\right)}$, set $X = \frac{V}{U}$, otherwise GOTO 1

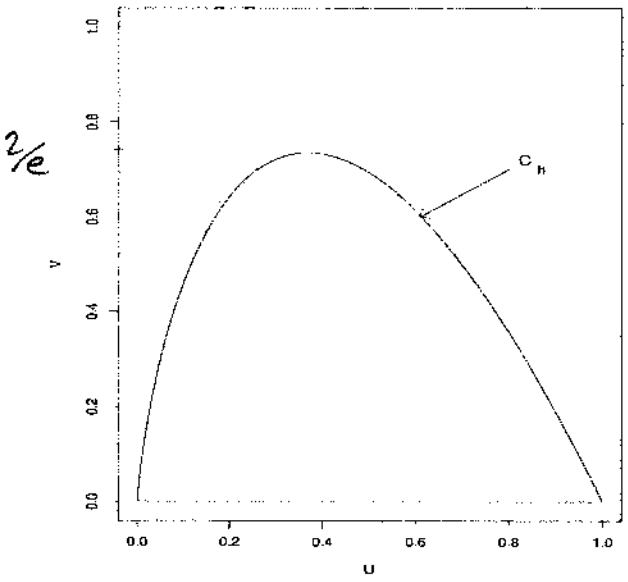
3.6.2 Acceptance Probability?

$$\begin{aligned} \text{Probability of accepting an } X &= \frac{\text{Area}(C_h)}{\text{Area of bounding rectangle}} \\ &= \frac{\frac{1}{2}\int h(x)\,\mathrm{d}x}{a(c-b)} \end{aligned}$$

Examples

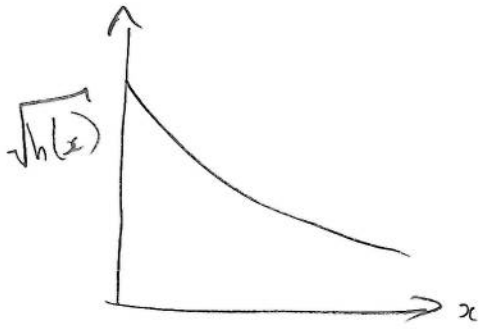
(i) $X \sim \text{Exp}(1)$, $h(x) = e^{-x}, x \geq 0$.

$$C_h = \{(u,v) : 0 \leq u \leq \sqrt{e^{-v/u}}\}.$$



$e^{-\frac{x}{2}}$

For a: Find $\sup_{x \geq 0} \sqrt{h(x)} := a$

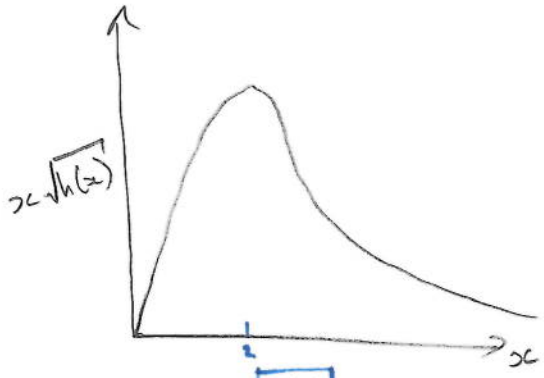


Is $h(x)$ bounded above in $x \geq 0$? ✓

$a = \sqrt{h(0)} = 1$.

For b, c:

Check: is $x^2 h(x)$ bounded for $x \geq 0$?



Supremum $x\sqrt{h(x)}$ over $x \geq 0$

$\frac{d}{dx}\left(xe^{-\frac{x}{2}}\right) = \left(1 - \frac{x}{2}\right)e^{-\frac{x}{2}} = 0 \Rightarrow x = 2$

$c := \sup_{x \geq 0} x\sqrt{h(x)} = 2e^{-1}$

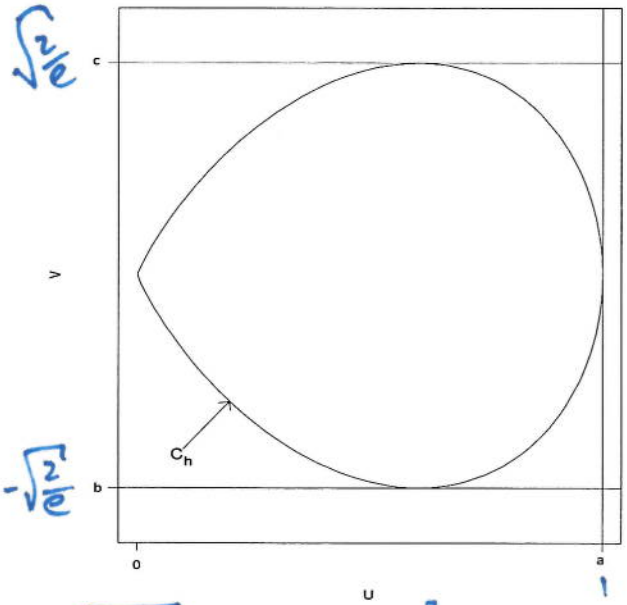
Minimum over $x \leq 0$ of $x\sqrt{h(x)} = 0\sqrt{h(0)} = 0 := b$

Algorithm

1. Generate $U_1, U_2 \sim U(0,1)$,
2. Set $U = U_1$; $V = 2U_2/e$
3. If $U \leq \sqrt{e^{-\frac{V}{U}}}$, ie $V \leq -2U \log U$, accept, set $X = \frac{V}{U}$
4. Otherwise GOTO 1.

(ii) $X \sim \mathrm{N}(0,1)$, $h(x) = e^{-x^2/2}$.

$$C_h = \{(u,v) : 0 \le u \le e^{\frac{-v^2}{4u^2}}\}.$$



For a: $\sup_x \sqrt{h(x)} = \sup_x e^{-\frac{x^2}{4}} = 1.$

For b, c: $x\sqrt{h(x)} = xe^{-\frac{x^2}{4}}$

$$\frac{d}{dx}\left(xe^{-\frac{x^2}{4}}\right) = \left(1-\frac{x^2}{2}\right)e^{-\frac{x^2}{4}} = 0 \Rightarrow x = \pm\sqrt{2}$$

$x = -\sqrt{2} \Rightarrow x\sqrt{h(x)} = -\sqrt{2/e} = b$

$x = +\sqrt{2} \Rightarrow x\sqrt{h(x)} = +\sqrt{2/e} = c$

Algorithm

1. Generate $U_1, U_2 \sim U(0,1)$
2. Set $U = U_1$; $V = -\sqrt{2/e} + 2\sqrt{\frac{2}{e}}U_2$
3. If $U \le e^{-\frac{V^2}{4U^2}}$, ie $V^2 \le -4U^2 \log U$, set $X = \frac{V}{U} \sim N(0,1)$
4. Otherwise GOTO 1.

Acceptance Probability?

$$\begin{aligned}
\text{P(accept an } X) &= \frac{\text{Area}(C_h)}{\text{Area of bounding rectangle}} \\
&= \frac{\frac{1}{2}\int h(x)\,\mathrm{d}x}{a(c-b)} \\
&= \frac{1}{2}\int e^{-\left(\frac{x^2}{2}\right)}\,dx \Big/ 2\sqrt{2}\,e^{-\frac{1}{2}} \\
&= \frac{\sqrt{2\pi}}{2} \times \frac{1}{2\sqrt{2}\,e^{-1/2}} = \frac{\sqrt{\pi e}}{4} \approx 0.731.
\end{aligned}$$

3.7 Pretesting–squeezing

Both the rejection and the ratio of uniforms methods use membership tests, e.g.

$$\begin{aligned}
MU &\leq \frac{f(x)}{g(x)} \\
(U,V) &\in C_h
\end{aligned}$$

Evaluation of the membership criterion may be computationally expensive, involving calls to trigonometric functions, logarithms, exponentials, etc; it would be beneficial to avoid such membership tests as much as possible.

Example - Rejection Sampling

Suppose we can find functions W_L and W_U which are inexpensive to calculate and are s.t.

$$W_L(x) \leq \frac{f(x)}{g(x)} \leq W_U(x), \forall x.$$

Then we could efficiently bypass our membership criterion for certain candidate values $y \sim g(\cdot)$:

$$\begin{aligned}
MU < W_L(y) &\Rightarrow \quad \text{set } X = y \\
MU > W_U(y) &\Rightarrow \quad \text{try again}
\end{aligned}$$

Specific Example

Suppose we wish to generate from the Exp(1) distribution truncated to (0,2). Our target density has the form

$$f(x) = \frac{e^{-x}}{\int_0^2 e^{-x}\,\mathrm{d}x} \quad x \in (0,2)$$

...in fact,

$$f(x) = \begin{cases} \frac{e^{-x}}{1-e^{-2}} & \text{for } x \in (0,2) \\ 0 & \text{otherwise} \end{cases}$$

Aside

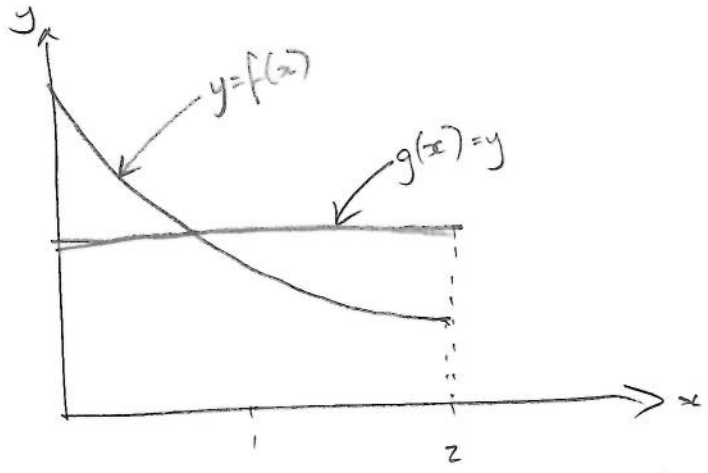
We could do this easily using inversion:

$$F(x) = \frac{1}{1-e^{-2}} \int_0^x e^{-s}\,\mathrm{d}s = \frac{1-e^{-x}}{1-e^{-2}}$$

So, $F^{-1}(U) = -\log[1 - U(1-e^{-2})]$.

Alternatively, we could simply generate from Exp(1) and then throw away points >2.

We consider a rejection algorithm to illustrate squeezing, and we take $U(0,2)$ as our instrumental (sampling) distribution.



$\frac{f(x)}{g(x)} = \frac{2e^{-x}}{1-e^{-2}}$ $\quad M = \sup_{x\in(0,2)} \frac{f(x)}{g(x)} = \frac{2}{1-e^{-2}} \approx 2.313\ldots$

Standard Rejection Algorithm

1. Generate $Y = y \sim U(0,2)$.
2. Generate $U = u \sim U(0,1)$.
3. If $Mu \leq \frac{f(y)}{g(y)}$ set X =y. , ie $u \leq e^{-y}$
   Otherwise GOTO 1.

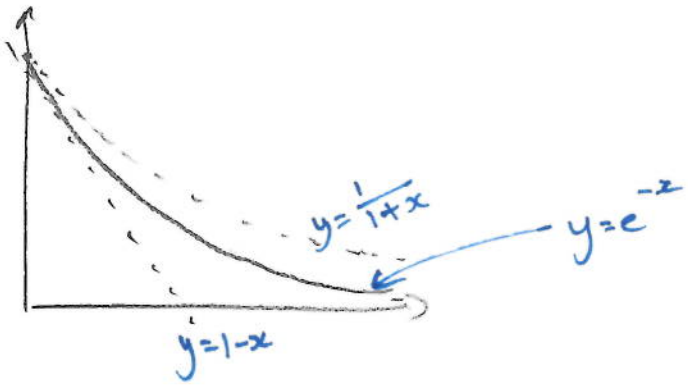
We now search for suitable functions $W_L(\cdot)$ and $W_U(\cdot)$:

$e^y \geq 1+y \quad \forall y$ (by Taylor).

$y=-x \Rightarrow e^{-x} \geq 1-x \quad \forall x$

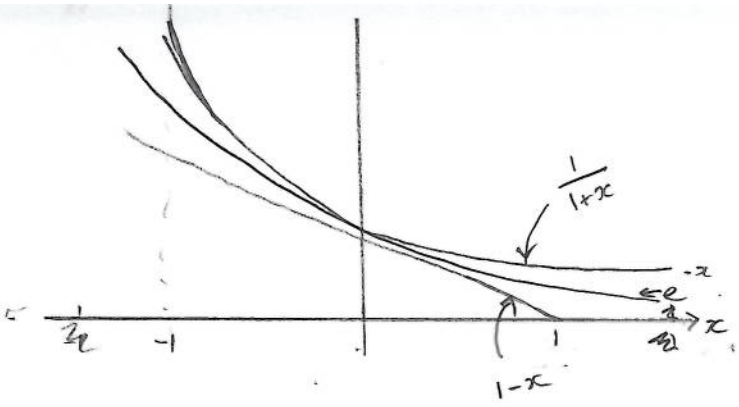
$e^x \geq 1+x \quad \forall x$

$\Rightarrow e^{-x} \leq \frac{1}{1+x} \quad \forall x > -1$



Algorithm - Rejection Sampling with Pre-squeezing

1. Generate $Y = y \sim U(0,2)$.
2. Generate $U = u \sim U(0,1)$.
3. If $u \leq 1-xy$, set $X=y$
4. If $u \geq \frac{1}{1+y}$, GOTO 1
5. Otherwise : if $u \leq e^{-y}$ set $X=y$ otherwise GOTO 1.



3.7.1 More refined squeezes

Recall:

$$1 - x \leq e^{-x} \leq \frac{1}{1+x} \quad \forall x > -1.$$

Can we get a 'tighter' squeeze on e^{-x}? Idea: use a shifted domain to tighten the squeeze.

Start with the left-hand inequality:

Write $x = y - a$, some $a \in \mathbb{R}$, though we have $a \leq 1$

$$e^{-x} \geq 1 - x \ \forall x \Rightarrow e^{-y} e^{a} \geq 1 - y + a$$

$$\Rightarrow e^{-y} \geq e^{-a}(1 - y + a) \ \forall y$$

Then, for the right-hand inequality,

Write $x = y - b$

$$e^{-y} e^{b} \leq \frac{1}{1+y-b} \Rightarrow e^{-y} \leq \frac{e^{-b}}{1+y-b} \quad \forall y > b - 1$$

Combining the two we have

$$e^{-a}(1-y+a) \leq e^{-y} \leq \frac{e^{-b}}{1+y-b} \quad \forall y > b-1$$

a and b are tuning parameters which we can use to make the squeezing more efficient.

Algorithm (for sampling from Exp(1) truncated to $(0, 2)$)

Setup: Choose optimal a & b; calculate & store e^{-a}, e^{-b}

1. Generate $Y = y \sim U(0, 2)$ and $U = u \sim U(0, 1)$.
2. If $u \leq e^{-a}(1-y+a)$, set $X = y$.
3. If $u \geq \frac{e^{-b}}{1+y-b}$, GOTO 1.
4. Otherwise, if $u \leq e^{-y}$, set $X = y$, o.w. GOTO 1.

Remark 1 We *choose* a and b so as to maximize the efficiency of the algorithm

ie we minimise $P\left[e^{-a}(1-y+a) < u < \frac{e^{-b}}{1+y-b}\right]$

Remark 2 We need only calculate e^{-a} and e^{-b} once

Example

Use a ratio-of-uniforms method with squeezing to generate $X \sim \text{Exp}(1)$.

Test condition (algorithm on p31):

$$V \leq -2U \log U$$

If we can find UB & LB for $-\log U$ we can set $W_L := u \times LB$, $W_U := u \times UB$

Idea: LB on e^x → UB on $\log(x)$ → LB on $-\log(x)$ & UB on $-\log(x)$

Using our previous lower bound for e^x,

$e^x \geq 1+x \ \forall x \Rightarrow x \geq \log(1+x) \quad x > -1$

Write $y = x+1 \Rightarrow y - 1 \geq \log y \Rightarrow 1 - y \leq -\log y \quad (\forall y > 0)$

Writing $z = \frac{1}{y} \Rightarrow \frac{1}{z} - 1 \geq \log \frac{1}{z} \Rightarrow \frac{1}{z} - 1 \geq -\log z \quad (\forall z > 0)$

Now, writing $y = au$ gives

$$1 - au \leq -\log u - \log a$$

$$1 - au + \log a \leq -\log u \qquad (3.1)$$

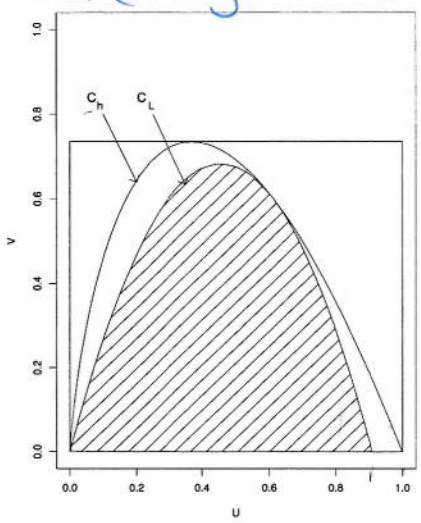
and writing $z = u/b$ gives

$$\frac{b}{u} - 1 \geq -\log u + \log b$$

$$\frac{b}{u} - 1 - \log b \geq -\log u \qquad (3.2)$$

Combining (3.1) & (3.2):

$$2u(1 + \log a - au) \leq -2u \log u \leq 2b - 2u - 2u \log b.$$



C_L is shown here for a particular value of a.

Algorithm

1. Generate $U_1, U_2 \sim U(0,1)$.
2. Set $U = U_1$, $V = 2U_2/e$.
3. If $V < 2U(1 + \log a - aU)$, set $X = V/U$.
4. If $V > 2b - 2U(1 + \log b)$ GOTO 1.
5. Otherwise: If $V < -2U \log U$, set $X = V/U$, o.w. GOTO 1.

Calculation of optimal a, b:

We wish to choose a that maximises the probability of acceptance at step 3 of the algorithm above; intuitively, this can be found by maximising the area of C_L in the figure on the previous page.

Formally, we wish to maximise

$$\mathrm{P}(\text{accept } X) = \int_0^1 \mathrm{P}\left[V < 2U(1 + \log a) - 2aU^2 \middle| U = u\right] f_U(u)du,$$

with $U \sim U(0,1)$ and $V \sim U(0, 2e^{-1})$. Note that, for some values of $a > 0$, we will have $2u(1 + \log a) - 2au^2 < 0$ for some possible values $u > u^* \in [0,1]$. Since $V \geq 0$, we therefore have that

$$\mathrm{P}\left[V < 2U(1+\log a) - 2aU^2 \middle| U = u\right] = 0 \text{ for } u > u^*$$

and so

$$\mathrm{P}(\text{accept } X) = \int_0^{u^*} \int_0^{2u(1+\log a) - 2au^2} f_V(v)\, dv\, f_U(u) du$$

$$= \frac{e}{2} \int_0^{u^*} \int_0^{2u(1+\log a) - 2au^2} dv\, du$$

$$= e\left[\frac{u^2}{2}(1 + \log a) - \frac{au^3}{3}\right]_0^{u^*}$$

Find u^*: $2u(1+\log a) - 2au^2 \geq 0$

$$\Rightarrow u < \frac{1+\log a}{a} := u^*$$

So

$$\mathrm{P}(\text{accept } X) \;=\; \frac{e(1+\log a)^3}{6a^2}$$

This probability can now be straightforwardly maximised as a function of a:

$$\log[\mathrm{P}(\text{accept } X)] = 1 + 3\log(1+\log a) - \log 6 - 2\log a$$

$$\frac{\mathrm{d}}{\mathrm{d}a}\log[\mathrm{P}(\text{accept } X)]\Big|_{a=\hat{a}} = 0 \;\Rightarrow\; \frac{3}{1+\log\hat{a}}\cdot\frac{1}{\hat{a}} - \frac{2}{\hat{a}} = 0$$
$$\Rightarrow\; \hat{a} = e^{1/2} \approx 1.65.$$

b can be found similarly by maximising the probability of rejecting at step 4 of the same algorithm.

It is found that $a \approx 1.65, b \approx 1.05$ maximises the efficiency of the squeezed ratio-of-uniforms method in this instance.

3.8 Random variate generation for discrete distributions

Recall: inversion method with

$$F^-(u) = \min\{x, F_X(x) \geq u\}$$

W.l.o.g. assume that discrete distribution is defined by

$$p_i = \mathrm{P}(X = i) \quad i = 1, 2, \ldots$$

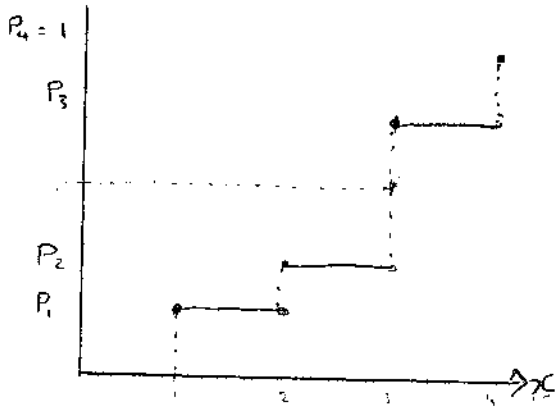
Let the number of points (values for X) be N, possibly infinite,

$$P_i = \mathrm{P}(X \leq i) = \sum_{j=1}^{i} p_j$$

Then

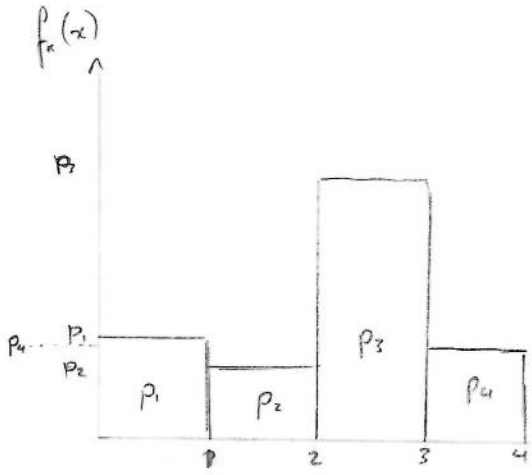
$$F^-(u) = i, \quad \text{where} \quad P_{i-1} < u \leq P_i$$

with $P_0 = 0$.



Algorithm

1. Set i=1.
2. Generate $U = u \sim U(0,1)$.
3. If $u \leq P_i$, set $X = i$.
4. Otherwise $i \to i+1$, GOTO 3.



This starts from the left, so to return $X = i$ from this algorithm requires i comparisons.

$$\text{E(Number of comparisons)} = \sum ip_i = \text{E}(X)$$

which can be very inefficient.

3.8.1 R–L Method

If the p.d.f. is known to be roughly "symmetric and unimodal", we could start the algorithm in the "middle" and iterate left and right.

Algorithm

1. Set $L = 0, R = N$.
2. Generate $U = u \sim U(0,1)$.
3. Set $i = \lfloor \frac{L+R}{2} \rfloor$.
4. $\begin{cases} \text{if } u > P_i & \text{set } L = i \\ \text{otherwise} & \text{set } R = i \end{cases}$
5. If $L \geq R - 1$, set $X = i$.
6. Otherwise, GOTO 3.

NB: Only 1 uniform generated

3.8.2 The "Table" method

This is actually a composition method. Recall we wish to generate from a discrete distribution with

$$p_i = \mathrm{P}(X = x_i) \quad i = 1, \ldots, N.$$

Write

$$p_i = \frac{a_{i1}}{10} + \frac{a_{i2}}{100} + \frac{a_{i3}}{1000} + \cdots = \sum_{j=1}^{d} \frac{a_{ij}}{10^j}$$

$$= \frac{\sum_1^N a_{i1}}{10} \frac{a_{i1}}{\sum_{i=1}^N a_{i1}} + \frac{\sum_1^N a_{i2}}{100} \frac{a_{i2}}{\sum_1^N a_{i2}} + \cdots + \frac{\sum_1^N a_{id}}{10^d} \frac{a_{id}}{\sum_1^N a_{id}}$$

where d = no. of decimal places taken

Example – Generate $X \sim \mathrm{Bin}(3, \frac{1}{3})$.

$$\begin{aligned} p_0 &= \mathrm{P}(X = 0) = 0.296 \\ p_1 &= \mathrm{P}(X = 1) = 0.445 \\ p_2 &= \mathrm{P}(X = 2) = 0.222 \\ p_3 &= \mathrm{P}(X = 3) = 0.037 \end{aligned}$$

$$\begin{aligned} p_0 &= 0.296 = 0.8 \times \tfrac{2}{8} + 0.18 \times \tfrac{9}{18} + 0.02 \times \tfrac{6}{20} \\ p_1 &= 0.445 = 0.8 \times \tfrac{4}{8} + 0.18 \times \tfrac{4}{18} + 0.02 \times \tfrac{5}{20} \\ p_2 &= 0.222 = 0.8 \times \tfrac{2}{8} + 0.18 \times \tfrac{2}{18} + 0.02 \times \tfrac{2}{20} \\ p_3 &= 0.037 = 0.8 \times \tfrac{0}{8} + 0.18 \times \tfrac{3}{18} + 0.02 \times \tfrac{7}{20} \end{aligned}$$

(0.8×2 ; 0.01×9 ; 0.001×6)

Algorithm

1. Generate $U = u \sim U(0, 1)$.

2. If $0 \leq u < 0.8$ then set

   $X = 0$ w.p. $\frac{2}{8}$, $X = 1$ w.p. $\frac{4}{8}$, $X = 2$ w.p. $\frac{2}{8}$, $X = 3$ w.p. 0;

   If $0.8 \leq u < 0.98$ then set

   $X = 0$ w.p. $\frac{9}{18}$, $X = 1$ w.p. $\frac{4}{18}$, $X = 2$ w.p. $\frac{2}{18}$, $X = 3$ w.p. $\frac{3}{18}$;

   If $0.98 \leq u < 1$ then set

   $X = 0$ w.p. $\frac{6}{20}$, $X = 1$ w.p. $\frac{5}{20}$, $X = 2$ w.p. $\frac{2}{20}$, $X = 3$ w.p. $\frac{7}{20}$.

Why does this work?

$$P(X=0) = \underbrace{0.8}_{\pi_1} \times \frac{2}{8} + \underbrace{0.18}_{\pi_2} \times \frac{9}{18} + \underbrace{0.02}_{\pi_3} \times \frac{6}{20}$$

(f_{01}, f_{02}, f_{03})

Recall the composition method:

$$f(x) = \pi_1 f_1(x) + \cdots + \pi_k f_k(x)$$

For discrete RVs, this becomes: $p_i = \pi_1 f_{i1} + \pi_2 f_{i2} + \pi_3 f_{i3} + \cdots + \pi_k f_{ik}$, $i = 1, \ldots, N$

Here, we have $\pi_1 = 0.8$

$\pi_2 = 0.18$

$\pi_3 = 0.02$

$$(f_{01}, f_{11}, f_{21}, f_{31}) = \left(\frac{2}{8}, \frac{4}{8}, \frac{2}{8}, 0\right)$$
$$(f_{02}, f_{12}, f_{22}, f_{32}) = \left(\frac{9}{18}, \frac{4}{18}, \frac{2}{18}, \frac{3}{18}\right)$$
$$(f_{03}, f_{13}, f_{23}, f_{33}) = \left(\frac{6}{20}, \frac{5}{20}, \frac{2}{20}, \frac{7}{20}\right).$$

Alternative Algorithm

1. Generate $U \sim U(0,1)$.

2. Select $j = j^*$ from

j	π_j
1	0.8
2	0.18
3	0.02

3. Generate $V \sim U(0,1)$. Given j^* generate $i = i^*$ from f_{ij^*} with

i	f_{ij^*}
0	f_{0j^*}
1	f_{1j^*}
2	f_{2j^*}
3	f_{3j^*}

4. Set $X = i^* \sim \text{Bin}(3, \frac{1}{3})$

A disadvantage to this method is the need to find suitable π_i's and f_{ij}'s.

3.8.3 Alias method

Lastly, we consider a composition method involving 'two-point' distributions.

Example – Generate $X \sim \text{Bin}(3, \frac{1}{3})$.

i	p_i	
0	$\left(\frac{2}{3}\right)^3$	$\frac{8}{27}$
1	$3\left(\frac{1}{3}\right)\left(\frac{2}{3}\right)^2$	$\frac{12}{27}$
2	$3\left(\frac{1}{3}\right)^2\left(\frac{2}{3}\right)$	$\frac{6}{27}$
3	$\left(\frac{1}{3}\right)^3$	$\frac{1}{27}$

We can write these probabilities as

$$\begin{aligned}
\tfrac{8}{27} &= \tfrac{1}{4}(\tfrac{9}{27} + 0 + 0 + \tfrac{23}{27}) \\
\tfrac{12}{27} &= \tfrac{1}{4}(\tfrac{18}{27} + \tfrac{9}{27} + \tfrac{21}{27} + 0) \\
\tfrac{6}{27} &= \tfrac{1}{4}(0 + \tfrac{18}{27} + \tfrac{6}{27} + 0) \\
\tfrac{1}{27} &= \tfrac{1}{4}(0 + 0 + 0 + \tfrac{4}{27})
\end{aligned}$$

Again, we have

$$p_i = \sum \underbrace{\pi_j}_{\frac{1}{4}} f_{ij}$$

With f's: $\begin{pmatrix} \frac{9}{27} \\ \frac{18}{27} \\ 0 \\ 0 \end{pmatrix}$ $\begin{pmatrix} 0 \\ \frac{9}{27} \\ \frac{18}{27} \\ 0 \end{pmatrix}$ $\begin{pmatrix} 0 \\ \frac{21}{27} \\ \frac{6}{27} \\ 0 \end{pmatrix}$ $\begin{pmatrix} \frac{23}{27} \\ 0 \\ 0 \\ \frac{4}{27} \end{pmatrix}$

The algorithm for this procedure is exactly as that for the Table Method.

Here, however, we note that the component distributions f_{ij} all only have two positive elements, so we will only need 1 comparison in sampling from f_{ij}

Q: Is there always going to be such a decomposition?

A: YES! This is Vose's method, this is a highly efficient alternative to the Table method.

4 Monte Carlo Integration

Suppose we wish to estimate the value of an integral

$$\theta = \int h(x)\,\mathrm{d}x$$

which is analytically intractable. NB: ∃ an exact answer - deterministic

Note: There are many numerical procedures for dealing with this, e.g. Simpson's rule, trapezium rule, ... we seek an alternative!

Suppose we write

$$\begin{aligned}\theta &= \int h(x)\,\mathrm{d}x \\ &= \int \phi(x) f(x)\,\mathrm{d}x \quad = E_f[\phi(X)]\end{aligned}$$

such that $f(\cdot)$ is a pdf and $\phi(\cdot) = \frac{h}{f}$.

This motivates a **probabilistic approach**!

Assuming we can generate $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} f(\cdot)$, we estimate θ using

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} \phi(X_i)$$

This is the Mean-Value (or "Crude") Monte Carlo Estimator.

What are the properties of this estimator?

Expectation...?

$$\begin{aligned}E_f(\hat{\theta}) &= E_f\left[\tfrac{1}{n}\sum_{i=1}^{n}\phi(X_i)\right] \\ &= \tfrac{1}{n}\, n\, E_f[\phi(X)] \\ &= \theta\end{aligned}$$

Variance...?

$$\begin{aligned}\mathrm{Var}(\hat{\theta}) &= \mathrm{Var}\left[\tfrac{1}{n}\sum_{i=1}^{n}\phi(X_i)\right] \\ &= \tfrac{1}{n^2}\, n\, \mathrm{Var}[\phi(X)] \\ &= \tfrac{1}{n} E\left[\{\phi(X) - E[\phi(X)]\}^2\right] \\ &= \tfrac{1}{n}\int [\phi(x) - \theta]^2 f(x)\,\mathrm{d}x \\ &= \frac{k}{n}\end{aligned}$$

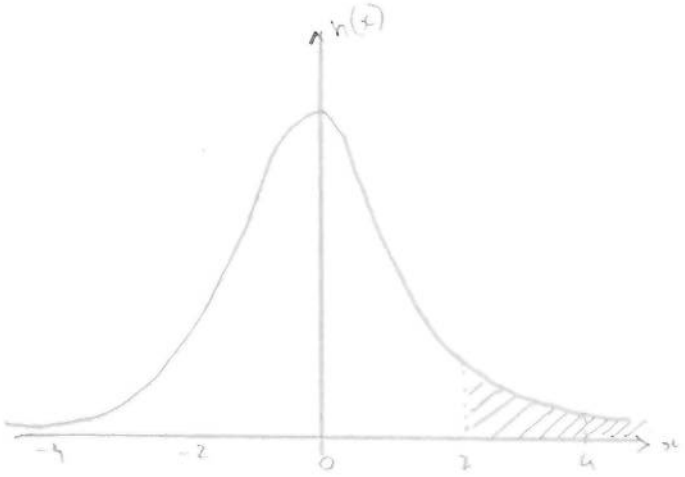
So the estimator is unbiased and large samples lead to more accurate estimators.

Can we control the accuracy of $\hat{\theta}$ for a sample of fixed size?

- For each $h(x)$, $\exists$ many different alternatives for (ϕ, f).
- Different choices of (ϕ, f) $\longrightarrow$ different value of k
- Want to choose (ϕ, f) such that k is small.

Example

$$\theta = \int_2^\infty \frac{1}{\pi(1+x^2)}\,\mathrm{d}x \qquad [= \mathrm{P}(X > 2) \text{ where } X \sim \text{Cauchy}]$$



We know $\theta \approx 0.1476$

Possible choices for the (ϕ, f) decomposition:

Since E_f is defined over the entire support of f, we want to limit the positive support of ϕf using ϕ.

(i)

$$f(x) = \frac{1}{\pi(1+x^2)}, \qquad \phi(x) = \mathbb{I}(x > 2) = \begin{cases} 1 & x > 2 \\ 0 & x \leq 2 \end{cases}$$

i.e. generate n Cauchy variates and count the number, Y, greater than 2.

$\hat{\theta} = \frac{Y}{n}$ $\qquad Y \sim \text{Bin}(n, \theta)$

$$\text{Var}[\hat{\theta}] = \frac{1}{n^2}\text{Var}[Y]$$

$$= \frac{1}{n^2} n\theta(1-\theta)$$

$$= \frac{0.126}{n}$$

(ii) Think of $\theta = \frac{1}{2}\mathrm{P}(|X| > 2)$.

$$f(x) = \frac{1}{\pi(1+x^2)}, \quad \phi(x) = \mathbb{I}(|x| > 2).$$

i.e. generate n Cauchy variates and count the number, Y, s.t. $|Y| > 2$.

$\hat{\theta} = \frac{1}{2}\frac{Y}{n}$ $\quad Y \sim \mathrm{Bin}(n, 2\theta)$.

$$\mathrm{Var}[\hat{\theta}] = \frac{1}{4n^2}\mathrm{Var}[Y]$$
$$= \frac{1}{4n^2} n 2\theta(1-2\theta)$$
$$= \frac{0.052}{n}$$

Reducing the variance by a factor of ~ 2.4.

(iii) Equivalently, define

$$\begin{aligned}\lambda &= 1 - 2\theta = \int_{-2}^{2} \frac{1}{\pi(1+x^2)}\,\mathrm{d}x \\ &= 2\int_{0}^{2} \frac{1}{\pi(1+x^2)}\,\mathrm{d}x = 2\mathrm{P}(0 < X < 2) \\ &= 2\lambda'\end{aligned}$$

To estimate $\lambda' = \displaystyle\int_0^2 \frac{1}{\pi(1+x^2)}\,\mathrm{d}x$, we use

$$\hat{\lambda}' = \frac{1}{n}\sum_{i=1}^{n} \frac{2}{\pi(1+X_i^2)}, \qquad X_i \overset{\text{i.i.d}}{\sim} U(0,2)$$

i.e.

$$\phi(x) = \frac{2}{\pi(1+x^2)}, \quad f(x) = \frac{1}{2} \text{ for } x \in (0,2).$$

$$\begin{aligned}\mathrm{var}(\hat{\lambda}') &= \frac{\mathrm{var}(\phi(X))}{n} \\ &= \frac{1}{n}\{\mathrm{E}(\phi^2(X)) - \mathrm{E}^2(\phi(X))\} \\ &= \frac{1}{n}\left\{\mathrm{E}(\phi^2(X)) - (\lambda')^2\right\} \\ &= \frac{1}{n}\left\{\frac{4}{\pi^2}\frac{1}{2}\int_0^2 \frac{1}{(1+x^2)^2}\,\mathrm{d}x - \left(\frac{1}{2} - 0.1476\right)^2\right\} \\ &= \frac{0.0285}{n}\end{aligned}$$

$\lambda' = \frac{1}{2} - \theta$

$= \mathrm{var}(\hat{\theta})$ ... a further reduction of $\simeq 1.8$.

(iv) Note that if $Y = \frac{1}{X}$

$$\begin{aligned}\theta &= \int_2^\infty \frac{1}{\pi(1+x^2)}\,\mathrm{d}x \\ &= \int_0^{\frac{1}{2}} \frac{y^{-2}}{\pi(1+y^{-2})}\,\mathrm{d}y \\ &= \int_0^{\frac{1}{2}} \frac{1}{\pi(1+y^2)}\,\mathrm{d}y\end{aligned}$$

Take $f(y) = 2$, ie $Y \sim U(0, \frac{1}{2})$.

$\Rightarrow \phi(y) = \frac{1}{2\pi(1+y^2)} \Rightarrow \hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2\pi(1+Y_i^2)}$

$\mathrm{Var}(\hat{\theta}) = \frac{1}{n}\left\{\int_0^{\frac{1}{2}} \frac{1}{2\pi^2(1+y^2)^2}\,\mathrm{d}y - \theta^2\right\} = \frac{9\times10^{-5}}{n}$

A further improvement of factor $\frac{0.0286}{9\times10^{-5}} > 300$

We now consider an alternative to Crude Monte Carlo: the "Hit or Miss" method.

Hit-or-Miss Monte-Carlo

Let h be a bounded function on (a, b), with $0 \le h \le c$. As before, we seek

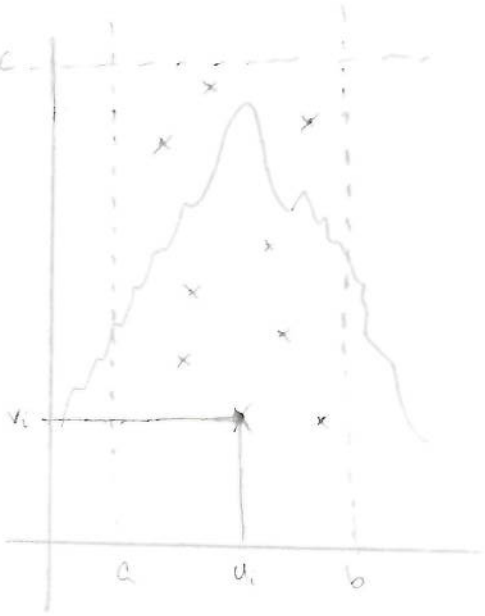
$$\begin{aligned}\theta &= \int_a^b h(x)\,\mathrm{d}x = \text{area under curve} \\ &= (b-a)\int_a^b h(x)f(x)\,\mathrm{d}x \quad \text{where } f(x) = \frac{1}{b-a}, \quad x \in (a,b)\end{aligned}$$

Approach:

Sample $U = u_i \sim U(a,b)$, $V = v_i \sim U(0,c)$, $i = 1, \ldots, n$

Define

$$\tilde{\theta} = \underbrace{c(b-a)}_{\text{area of rectangle}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbb{I}(v_i \le h(u_i))}_{\text{frequency of points falling under } h.}$$



Now, we consider the properties of $\tilde{\theta}$:

Expectation... $\mathbb{E}[\tilde{\theta}] = \frac{c(b-a)}{n} n \mathbb{E}\left[\mathbb{I}(V \le h(U))\right]$

$\mathbb{E}[\tilde{\theta}] = c(b-a)P(V \le h(U))$ $\sim$ Bernoulli $(P(V \le h(U)))$

$= \theta$

$X \sim$ Bernoulli (p)

$\mathbb{E}[X] = p$

Variance... $\mathrm{Var}[X] = p(1-p)$

$$\mathrm{var}(\tilde{\theta}) = \frac{c^2(b-a)^2}{n^2} n \,\mathrm{Var}\left[\mathbb{I}(V \le h(U))\right]$$

$$= \frac{c^2(b-a)^2}{n}\left[\frac{\theta}{c(b-a)}\left(1 - \frac{\theta}{c(b-a)}\right)\right]$$

$$= \frac{\theta}{n}\left[c(b-a) - \theta\right]$$

The question is...how does this compare to Crude Monte Carlo?

Recall: (for uniform f and suitable corresponding ϕ)

Choose

$$f(x) = \frac{1}{b-a}, \quad \phi(x) = (b-a)h(x), \qquad \Rightarrow \qquad \theta = \int_a^b h(x)\,\mathrm{d}x = \int_a^b \phi(x) f(x)\,\mathrm{d}x,$$

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} \phi(X_i), \qquad X_i \sim U(a,b)$$

$$= (b-a)\frac{1}{n}\sum_{i=1}^{n} h(X_i).$$

We know that $\mathrm{E}(\hat{\theta}) = \theta, \qquad \mathrm{var}(\hat{\theta}) = \frac{1}{n}\int_a^b [\phi(x) - \theta]\, f(x) dx$

$$\Rightarrow \mathrm{var}(\hat{\theta}) = \frac{(b-a)^2}{n}\int_a^b \left[h(x) - \frac{\theta}{b-a}\right]^2 f(x)\,\mathrm{d}x$$

$$= \frac{b-a}{n}\left\{\int_a^b h^2(x)\,dx - \frac{\theta^2}{b-a}\right\}$$

$$\le \frac{b-a}{n}\left\{c\int_a^b h(x)\,dx - \frac{\theta^2}{b-a}\right\} = \mathrm{var}\left[\tilde{\theta}\right]$$

Crude Monte Carlo always has smaller variance than Hit-or-Miss

$\Rightarrow$ NEVER use Hit or Miss

4.1 Importance Sampling

Crude Monte Carlo requires the ability to sample from the density $f(x)$...what if direct sampling isn't possible?

Suppose we can sample instead from an auxiliary density, $g(x)$, and we can pointwise evaluate both $f(x)$ and $g(x)$. We can write

$$\theta = \int \phi(x) f(x)\,\mathrm{d}x = \int \phi(x) \frac{f(x)}{g(x)} g(x)\,dx$$

$$= \int \phi(x)\, W(x) g(x) dx = E_g\left[W(x)\,\phi(x)\right]$$

Necessary condition : g must dominate f ; $f(x) > 0 \Rightarrow g(x) > 0$

Now, given $X_1, \ldots, X_n \overset{iid}{\sim} g(\cdot)$, we can estimate θ using a weighted sample mean:

$$\hat{\theta}_{IS} = \frac{1}{n} \sum_{i=1}^{n} W(X_i)\phi(X_i).$$

W is a function that ascribes an 'importance weight' to each variate $X_i \sim g$; it quantifies the importance of X_i in being used to perform inference with respect to the 'target' distribution f.

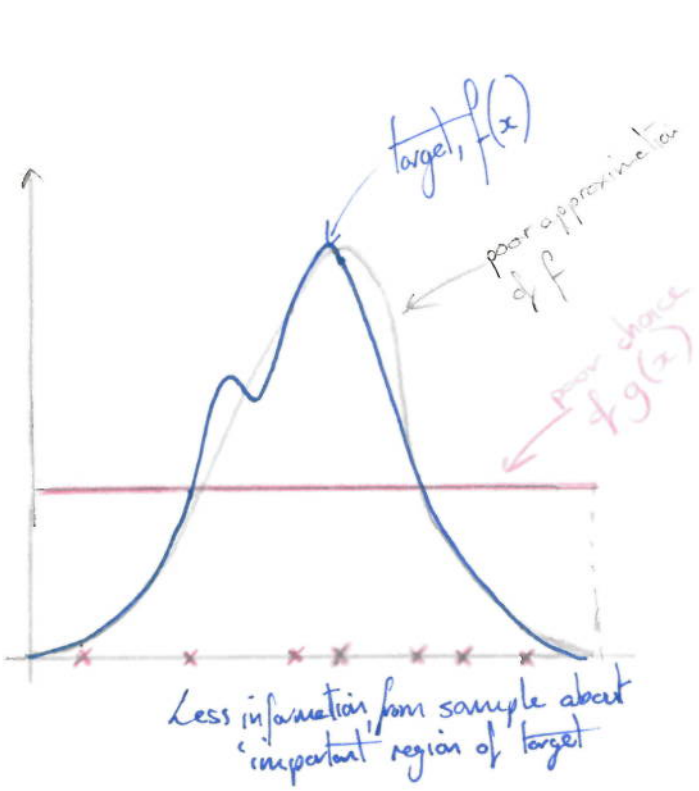
What are the properties of the estimator $\hat{\theta}_{IS}$?

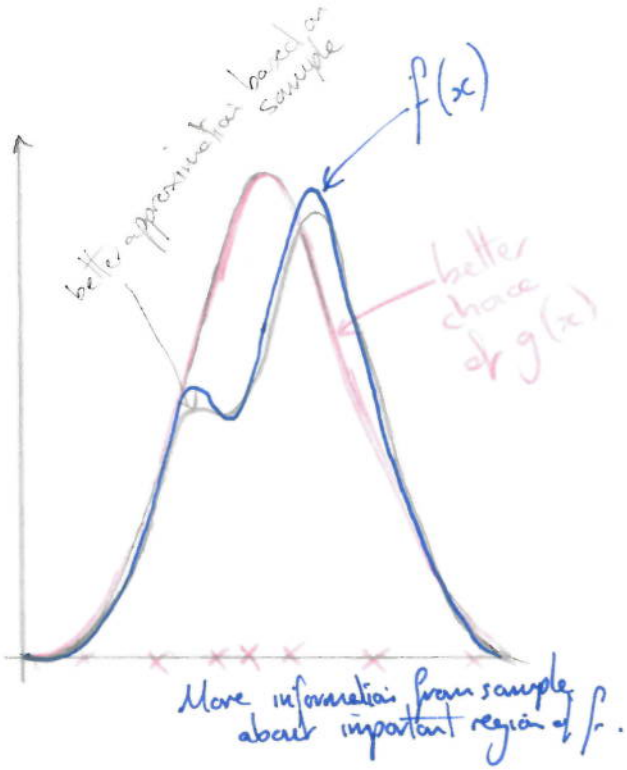
- As long as f and g are pointwise evaluated exactly, i.e. not up to a constant of proportionality, $\hat{\theta}_{IS}$ will be unbiased for θ. ie $E_g\left[\hat{\Theta}_{IS}\right] = \Theta$
- Variance...

$$\mathrm{Var}_g\left[\hat{\Theta}_{IS}\right] = \frac{1}{n^2} n \,\mathrm{Var}_g\left[W(X)\phi(X)\right]$$

$$= \frac{1}{n} \mathrm{Var}_g\left[\frac{f(X)}{g(X)}\phi(X)\right]$$

The variance is heavily dependent on our choice of g.
Ideally, we'd have g closely approximating ϕf, but for greater flexibility in testing different $\phi(x)$, we seek g that closely resembles f.





Example

$$\theta = \int_2^\infty \frac{1}{\pi(1+x^2)}\,\mathrm{d}x = \int_{-\infty}^{\infty} \mathbb{I}(x>2)\,\frac{1}{\pi(1+x^2)}\,f(x)\,\mathrm{d}x = \int_{-\infty}^{\infty} \phi(x)\,f(x)\,\mathrm{d}x$$

$g(x) = \frac{2}{x^2}$ is a pdf which mimics the "shape" of the integrand on $(2,\infty)$.

Can directly sample from g using inversion:

$$G(x) = \int_2^x \frac{2}{t^2}\,\mathrm{d}t = [-2t^{-1}]_2^x = 1 - \frac{2}{x}.$$

If $U \sim U(0,1)$ then $1-U \sim U(0,1)$, $\Rightarrow X = \frac{2}{U}$ is a random variable with pdf g.

We will show that this choice is equivalent to (iv) on page 47 - ie it gives the same variance.

$$\theta = \int_2^\infty \frac{1}{\pi(1+x^2)}\,\mathrm{d}x = \int_{-\infty}^{\infty} \underbrace{\mathbb{I}(x>2)}_{\phi(x)}\,\underbrace{\frac{x^2}{2\pi(1+x^2)}}_{f(x)/g(x)}\,\underbrace{\frac{2}{x^2}}_{g(x)}\,\mathrm{d}x$$

So,

$$\hat{\theta}_{IS} = \frac{1}{n}\sum_{i=1}^{n} W(X_i)\phi(X_i),$$

where,

$$X_1,\ldots,X_n \sim g(\cdot), \qquad W(x) = \frac{x^2}{2\pi(1+x^2)} \qquad \text{and} \qquad \phi(x) = \mathbb{I}(x>2).$$

$$\text{Var}_g\left[\hat{\theta}_{IS}\right] = \frac{1}{n}\text{Var}_g\left[w(x)\phi(x)\right]$$
$$= \frac{1}{n}\left\{E_g\left[w^2(x)\phi^2(x)\right] - \theta^2\right\}$$
$$= \frac{1}{n}\left\{\int_2^\infty \left[\frac{x^2}{2\pi(1+x^2)}\right]^2 \frac{2}{x^2}\,dx - \theta^2\right\}$$
$$= \frac{1}{n}\left\{\int_0^{\frac{1}{2}} \frac{1}{2\pi^2(1+y^2)^2}\,dy - \theta^2\right\} \qquad y = \frac{1}{x}$$
= variance from part (iv) on p47.

4.2 Antithetic Variates

Suppose $\hat{\theta}_1$ and $\hat{\theta}_2$ are both unbiased estimators of θ with variances $\text{var}(\hat{\theta}_1), \text{var}(\hat{\theta}_2)$.

$$\text{E}\left[\frac{1}{2}(\hat{\theta}_1 + \hat{\theta}_2)\right] = \theta \qquad \text{var}\left[\frac{1}{2}(\hat{\theta}_1 + \hat{\theta}_2)\right] = \frac{1}{4}\text{var}(\hat{\theta}_1) + \frac{1}{4}\text{var}(\hat{\theta}_2) + \frac{1}{2}\text{cov}(\hat{\theta}_1, \hat{\theta}_2).$$

Suppose $\text{var}(\hat{\theta}_1) = \text{var}(\hat{\theta}_2)$.

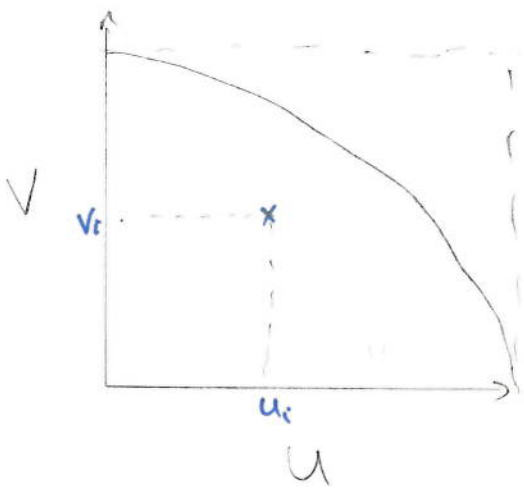
$$\Rightarrow \text{var}\left[\frac{1}{2}(\hat{\theta}_1 + \hat{\theta}_2)\right] = \frac{1}{2}\text{var}(\hat{\theta}_1) + \frac{1}{2}\text{cov}(\hat{\theta}_1, \hat{\theta}_2)$$
$$= \frac{1}{2}\text{var}(\hat{\theta}_1)\left[1 + \frac{\text{cov}(\hat{\theta}_1, \hat{\theta}_2)}{\sqrt{\text{var}(\hat{\theta}_1)\text{var}(\hat{\theta}_2)}}\right]$$
$$= \frac{1}{2}\text{var}(\hat{\theta}_1)\left[1 + \text{corr}(\hat{\theta}_1, \hat{\theta}_2)\right]$$

If $\text{corr}(\hat{\theta}_1, \hat{\theta}_2)$ is large and negative (ie close to -1), $\Rightarrow \text{var}\left[\frac{1}{2}(\hat{\theta}_1 + \hat{\theta}_2)\right] \ll \text{var}\left[\hat{\theta}_1\right]$

Example

$$\theta = \int_0^1 \sqrt{1-x^2}\,dx \quad \left(= \frac{\pi}{4}\right)$$

Hit-or-Miss MC



Generate $U = u_i \sim U(0,1)$, $V = v_i \sim U(0,1)$.

Let X = number of (u_i, v_i) such that $v_i < \sqrt{1-u_i^2}$ and set $\tilde{\theta} = \frac{X}{n}$.

$$X = n\tilde{\theta} \sim \text{Bin}\left(n, \frac{\pi}{4}\right)$$
$$\Rightarrow \text{var}(\tilde{\theta}) = \frac{\pi}{4n}\left(1 - \frac{\pi}{4}\right) \approx \frac{0.182}{n}$$

- Sampled $2n$ uniform variates.

Crude MC

Sample $X_1, \ldots, X_n$ from $f \sim U(0,1)$, and use $\phi(x) = \sqrt{1-x^2}$. Then,

$$\hat{\theta} = \frac{1}{n}\sum_i \phi(X_i)$$

$$\begin{aligned}
\text{var}(\hat{\theta}) &= \frac{1}{n}\text{var}(\phi(X)) \\
&= \frac{1}{n}\left\{\text{E}(\phi^2(X)) - \text{E}^2(\phi(X))\right\} \\
&= \frac{1}{n}\left[\int_0^1 \phi^2(x)\,\text{d}x - \theta^2\right] \\
&= \frac{1}{n}\left[\int_0^1 (1-x^2)\,\text{d}x - \left(\frac{\pi}{4}\right)^2\right] \\
&= \frac{1}{n}\left(\frac{2}{3} - \frac{\pi^2}{16}\right) \\
&= \frac{0.0498}{n}
\end{aligned}$$

- Sampling n uniforms $\Rightarrow$ $\frac{0.0249}{n}$ for 2n uniforms

$\Rightarrow$ an improvement by a factor $\frac{0.182}{0.0249} \approx 7$

Antithetic Version

Use uniforms twice – common way of doing this is by using U and $1-U$:

$$\phi(U) = \sqrt{1-U^2} \qquad , \phi(1-U) = \sqrt{1-(1-U)^2}$$

$$\Rightarrow \hat{\theta}_1 = \frac{1}{n}\sum\sqrt{1-U_i^2}, \qquad \hat{\theta}_2 = \frac{1}{n}\sum\sqrt{1-(1-U_i)^2}$$

So, if we set

$$\hat{\theta}^* = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\left[\sqrt{1-U_i^2} + \sqrt{1-(1-U_i)^2}\right] = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\left\{\phi(U_i)+\phi(1-U_i)\right\}$$

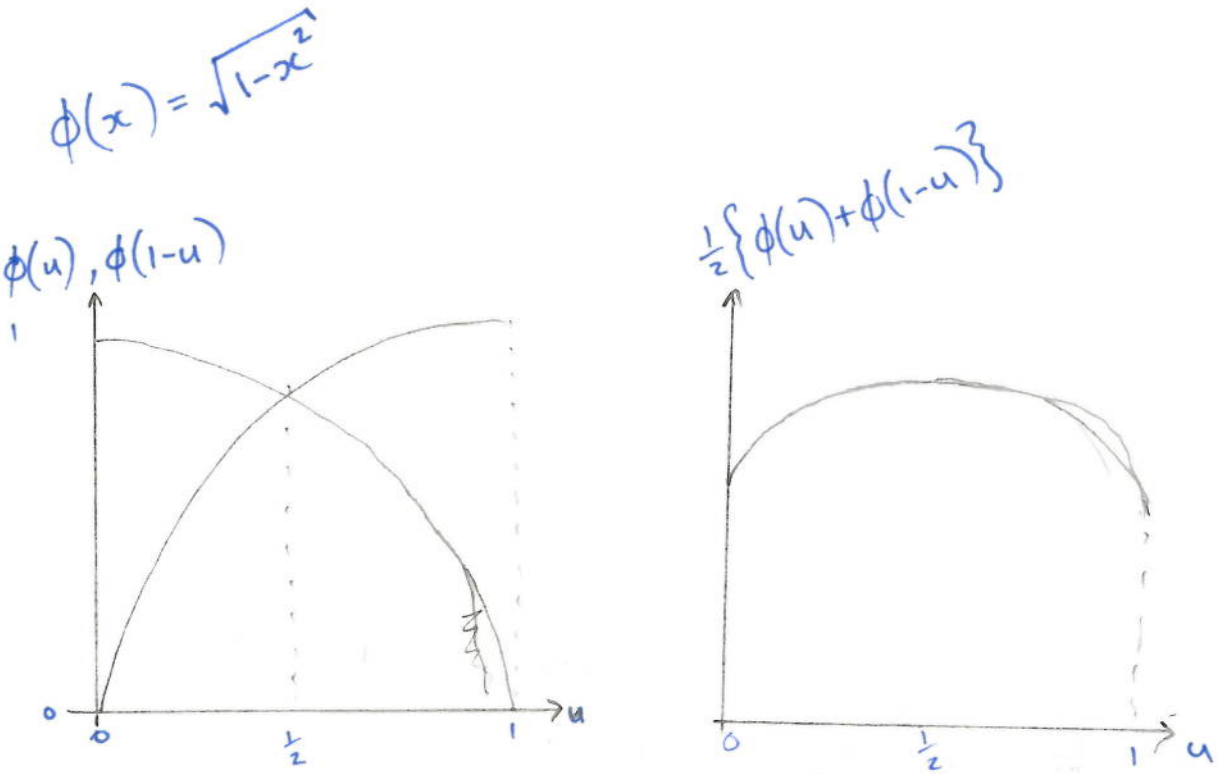
then

$$\begin{aligned}
\text{var}(\hat{\theta}^*) &= \frac{\text{var}(\hat{\theta}_1)}{2}(1+\text{corr}(\hat{\theta}_1,\hat{\theta}_2)) \\
&= \frac{1}{2n}\left(\frac{2}{3} - \frac{\pi^2}{16}\right)\left(1 + \underbrace{\text{corr}\left(\sqrt{1-U^2}, \sqrt{1-(1-U)^2}\right)}_{\text{Morgan p.165}}\right) \\
&= \frac{0.0052}{n}.
\end{aligned}$$

– 'Elements of Simulation' BJT Morgan (1984).

This gives a big reduction in variance!

Sampling n uniforms, evaluating ϕ twice for each, 🙂



The area under all the curves is $\frac{\pi}{4}$, but the ranges are 1 and $(\sqrt{3}-1)/2$, respectively. This reduction in the range leads to a reduction in variability.

General Theorem

Let g be a monotonic function on [0,1] and $U \sim U(0,1)$. Then

$$\text{corr}(g(U), g(1-U)) < 0$$

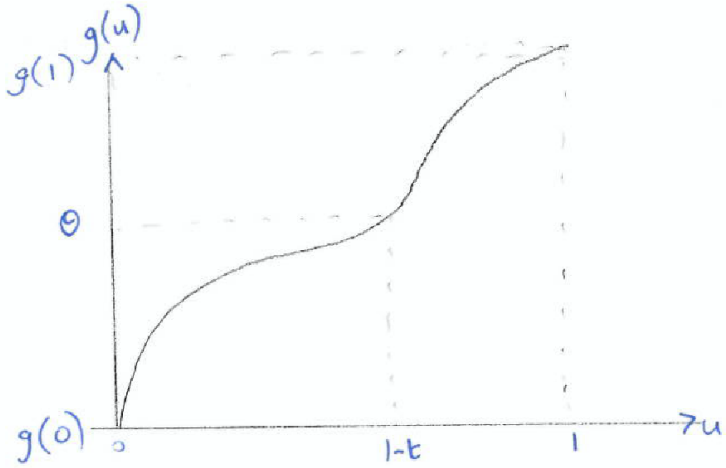
Proof

Wlog assume g is increasing. Let

$$\theta = \int_0^1 g(x)\,\mathrm{d}x = \mathrm{E}[g(U)] = \mathrm{E}[g(1-U)]$$

and

$$\inf\{u : g(u) > \theta\} = 1 - t$$



We have $\mathbb{E}[g(u)] = \mathbb{E}[g(1-u)] = \theta$

$$\mathrm{Cov}[g(u), g(1-u)] = \mathbb{E}[(g(u)-\theta)(g(1-u)-\theta)]$$

$$= \mathbb{E}[g(u)\{g(1-u)-\theta\}] - \theta\,\mathbb{E}[g(u)-\theta]$$

$$= \int_0^1 g(u)\{g(1-u)-\theta\}\,du$$

$u \in [0,t] \Rightarrow 1-u \in [1-t,1]$

$\Rightarrow g(1-u) > \theta$

$$= \int_0^t g(u)\{g(1-u)-\theta\}\,du + \int_t^1 g(u)\{g(1-u)-\theta\}\,du$$

$$< g(t)\int_0^t g(1-u)-\theta\,du \qquad < g(t)\int_t^1 g(1-u)-\theta\,du$$

as g is monotonic increasing & $\{g(1-u)-\theta\} > 0$

$$< g(t)\int_0^1 g(1-u)-\theta\,du = 0$$

So, if we want to evaluate

$$\theta = \int_0^1 \phi(u) f(u)\,\mathrm{d}u$$

where $f \sim U(0,1)$ and ϕ is monotonic then we can always use the estimator

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\{\phi(U_i) + \phi(1-U_i)\}$$

4.3 Control Variates

Once again, we consider the problem of estimating

$$\theta = \mathrm{E}_f[\phi(X)] \quad = \quad \mathbb{E}_f[Z]$$

via Monte Carlo integration. For notational convenience, write $Z = \phi(X)$.

Suppose $\exists W = \psi(X)$ such that $\mathrm{E}(W)$ is known and W is correlated with Z.

Then, given a sample $X_1, \ldots, X_n \overset{iid}{\sim} f$, we construct the following estimator:

$$\hat{\theta} \quad = \quad \frac{1}{n}\sum_{i=1}^{n}[Z_i - (W_i - \mathrm{E}(W))]$$

What are the properties of this estimator?

Expectation...

$$\mathbb{E}[\hat{\theta}] = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_f[\phi(X)] - \{\mathbb{E}[W] - \mathbb{E}[W]\} = \theta$$

so unbiased

Variance...

$$\begin{aligned}
\mathrm{Var}[\hat{\theta}] &= \frac{1}{n^2}\sum_{i=1}^{n}\mathrm{var}[Z_i - (W_i - \mathbb{E}[W])] \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\{\mathrm{var}(Z_i) + \mathrm{var}(W_i - \mathbb{E}[W]) - 2\mathrm{cov}(Z_i, W_i - \mathbb{E}[W_i])\} \\
&= \frac{1}{n}\{\mathrm{var}[Z] + \mathrm{Var}[W] - 2\mathrm{cov}(Z, W)\}
\end{aligned}$$

So, if we can choose W such that $\mathrm{cov}(Z, W)$ is high, then $\mathrm{var}(\hat{\theta})$ will be small.

Extension: We can construct a more general version of this estimator:

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n}[Z_i - \beta(W_i - \mathrm{E}(W_i))]$$

$$\Rightarrow \begin{cases} \mathbb{E}[\hat{\theta}] = \theta \quad \text{(unbiased)} \\ \mathrm{var}[\hat{\theta}] = \frac{1}{n}\{\mathrm{var}[Z] + \beta^2\mathrm{var}[W] - 2\beta\mathrm{cov}(Z, W)\} \end{cases}$$

Now, we can minimize the variance through optimal choice of β:

$$\frac{\mathrm{dvar}(\hat{\theta})}{\mathrm{d}\beta} = 0 \quad \Rightarrow \quad \frac{1}{n}\left\{-2\mathrm{cov}(Z, W) + 2\hat{\beta}\mathrm{var}(W)\right\} = 0 \qquad \Rightarrow \hat{\beta} = \frac{\mathrm{cov}(Z, W)}{\mathrm{var}(W)},$$

giving

$$\mathrm{var}(\hat{\theta}) = \frac{1}{n}\left\{\mathrm{var}(Z) - \frac{\mathrm{cov}^2(Z, W)}{\mathrm{var}(W)}\right\}$$

Unfortunately...in practice it is unlikely that we would know $\text{cov}(Z, W)$ and not $\text{E}(Z)$!

Solution: carry out a pilot study by generating $X_1, \ldots, X_n \sim f(\cdot)$ and then calculate

$$Z_i = \phi(X_i) \quad W_i = \psi(X_i)$$

Use sample to obtain estimate of β:

$$\hat{\beta} = \frac{\widehat{\text{cov}(Z, W)}}{\widehat{\text{var}(W)}} = \frac{\sum(Z_i - \bar{Z})(W_i - \text{E}(W_i))}{\sum(W_i - \text{E}(W_i))^2}.$$

Then use estimate of β in full study.

Further Extension: Suppose we have p control variates:

$$W_1, W_2, \ldots, W_p$$

such that

$$W_j = \psi_j(X) \quad \text{with} \quad \text{E}(W_j) \text{ known}$$

Then we can use all control variates to construct the estimator:

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n}\left[Z_i - \sum_{j=1}^{p}\beta_j(W_{ji} - \text{E}(W_j))\right]$$

$$\text{var}\left[\hat{\theta}\right] = \frac{1}{n}\text{var}\left[Z - \sum_{j=1}^{p}\beta_j\left(W_j - \mathbb{E}[W_j]\right)\right]$$

$$= \frac{1}{n}\left\{\mathbb{E}\left[\left(Z - \sum_{j=1}^{p}\beta_j\left(W_j - \mathbb{E}[W_j]\right)\right)^2\right] - \theta^2\right\}$$

minimizing variance $\Leftrightarrow$ minimizing $\text{E}\left[(Z - \sum_{j=1}^{p}\beta_j(W_j - \text{E}(W_j))^2\right]$.

We therefore use a pilot study to obtain estimates of $\underline{\beta} = (\beta_1, \ldots, \beta_p)$

$\hat{\underline{\beta}}$ will be the least squares estimator for $\underline{\beta}$!

(... this technique historically (occasionally) referred to as "regression sampling").

Example:

$$\begin{aligned}\theta &= \mathrm{P}(X > 2), \quad X \sim \text{Cauchy} \\ &= \int_2^\infty \frac{1}{\pi(1+x^2)}\,\mathrm{d}x\end{aligned}$$

Recall, version (iv) reduced to

$$\theta = \int_0^{\frac{1}{2}} \frac{1}{\pi(1+x^2)}\,\mathrm{d}x = \mathrm{E}_{U(0,\frac{1}{2})}\left[\frac{1}{2\pi(1+x^2)}\right] = \mathrm{E}_f\left[\phi(x)\right] = E_f[z].$$

How do we find suitable control variates?

$Z = \phi(X) = \frac{1}{2\pi(1+X^2)}$

$2\pi\phi(x) = \frac{1}{1+x^2} = 1 - x^2 + x^4 - \cdots$

So can use $W_1 = -X^2$ and $W_2 = -X^4$ as our control variates.

$$\mathbb{E}[W_1] = -\int_0^{\frac{1}{2}} 2x^2\,dx = -\frac{1}{12}$$

$$\mathbb{E}[W_2] = -\int_0^{\frac{1}{2}} 2x^4\,dx = -\frac{1}{80}$$

So, estimator becomes

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{1}{2\pi(1+X_i^2)} - \hat{\beta}_1\left(-X_i^2 + \frac{1}{12}\right) - \hat{\beta}_2\left(-X_i^4 + \frac{1}{80}\right)\right\}$$

A particular pilot study gave least squares estimates of $\hat{\beta}_1$ and $\hat{\beta}_2$ as

$$\hat{\beta}_1 = 0.1559, \quad \hat{\beta}_2 = -0.1166,$$

and a corresponding estimate of the mean squared residual:

$$\hat{\mathrm{E}}\left[\left(Z - \sum_j \beta_j(W_j - \mathrm{E}(W_j))\right)^2\right] = 0.0218$$

Since we also know that $\theta = 0.1476$, we can obtain:

$$\mathrm{var}(\hat{\theta}) \approx \frac{1}{n}\left[0.0218 - 0.1476^2\right] = \frac{1}{n} \times 9.1 \times 10^{-10}$$

when everything is calculated accurately.

→ taking $n = 1$ will do!

5 Generating Dependent Random Variables

Up to now, we have considered statistical problems that can be approached through simulating independent random variables from known distributions. Of course, this is not always the case - some statistical problems of interest require the practitioner to generate samples that exhibit some sort of dependence structure.

In this chapter, we consider techniques for generating random samples that display specific types of dependence - these can often be extended and applied to a number of different problems.

5.1 Multivariate Normal Distribution

Suppose we wish to generate

$$\underline{X} = (X_1, \ldots, X_p)^T$$

with $\underline{X} \sim \text{MVN}_p(\underline{\mu}, \Sigma)$ i.e.

$$f_X(\underline{x}) = (2\pi)^{-\frac{p}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\underline{x} - \underline{\mu})^{\mathrm{T}} \Sigma^{-1} (\underline{x} - \underline{\mu})\right]$$

where,

$$\underline{\mu} = (\mu_1, \ldots, \mu_p)^T, \qquad \Sigma_{p\times p} = (\Sigma_{ij})$$

$\mu_i = \mathbb{E}[X_i]$ $\qquad$ $\Sigma_{ii} = \text{Var}[X_i]$

$\Sigma_{\substack{ij \\ i\neq j}} = \text{Cov}[X_i, X_j]$

In order to generate $X_1, \ldots, X_p$, first generate $Z_i \sim \text{N}(0,1)$, $i = 1, \ldots, p$. Since Σ is positive definite, we can write $\Sigma = LL^{\mathrm{T}}$ - easy to find using the Cholesky decomposition, chol(.) in R.

Now, we can straightforwardly construct:

$$\underline{X} = \underline{\mu} + L\underline{Z}$$

with the required properties.

5.2 Poisson Process

Recall that a one-dimensional Poisson point process is used to model the occurrence of a number of events in an interval of time:

The prob. density for an event occurring at time t is $\lambda(t)$.

The homogeneous Poisson process is parameterised by the intensity, λ. If N_τ is the number of events in a period of length τ,

$$N_\tau \sim \text{Po}(\lambda\tau) \implies \mathbb{E}[N_\tau] = \text{Var}[N_\tau] = \lambda\tau$$

and these N_τ points are positioned uniformly within the interval of length τ. Thus, in order to sample a homogeneous Poisson process of rate λ over the interval (a, b), we simply draw $N \sim \text{Po}(\lambda(b-a))$ and then simulate $X_1, \ldots, X_N \sim U(a, b)$.

We also note that the time between consecutive events, T say, can be modelled as an Exponential RV

$$T \overset{iid}{\sim} \text{Exp}(\lambda) \implies \begin{cases} \mathbb{E}[T] = 1/\lambda \\ \text{Var}[T] = 1/\lambda^2 \end{cases}$$

Inhomogeneous Poisson processes can be generated using one of two methods. Suppose the intensity is a function of time t, $\lambda(t)$, and suppose we wish to generate an inhomogeneous Poisson process over (a, b). One can:

- split the interval (a, b) into M sub-intervals and approximate $\lambda(t)$ as piecewise constant, ie consider constant λ_m in subintervals $m = 1, \cdots, M$.
- or use a rejection-sampling procedure, drawing candidate values

  $$X_i = x \sim U(a, b)$$

  and accepting with probability

  $$\frac{\lambda(x)}{\int_a^b \lambda(y)dy}.$$

5.3 Order Statistics

It is often of interest to be able to generate ordered samples from known distributions.

Notation:

$$X_1, \ldots, X_n \quad \xrightarrow{\text{order}} \quad \underbrace{\overbrace{X_{(1)}}^{\text{MIN}} < \ldots < \overbrace{X_{(n)}}^{\text{MAX}}}_{\text{Order Statistics}}$$

INDEPENDENT $\xrightarrow{\text{order}}$ DEPENDENCE.

Surely we can just generate $X_1, \ldots, X_n$ and then order them?

Sorting algorithms are relatively expensive - useful to avoid them.

Idea: If we know $F_X(x)$, and we have access to a sample of ordered $U(0,1)$ variates

$$U_{(1)} < \ldots < U_{(n)},$$

we can set $X_{(i)} = F^{-1}(U_{(i)})$

This still involves ordering the $\{U_i\}$...fortunately, there are "tricks" to accomplish this!

(a) The Rescaling Method:

Generate $U_1, \ldots, U_n$ independent $U(0,1)$'s. Then iteratively define

$$\begin{aligned} U_{(n)} &= U_n^{\frac{1}{n}} \\ U_{(k)} &= U_{(k+1)} \times (U_k)^{\frac{1}{k}} \qquad k = n-1, n-2, \ldots, 1 \end{aligned}$$

Then $U_{(1)}, \ldots, U_{(n)}$ are ordered $U(0,1)$'s.

Proof:

Consider the cdf of $U_{(n)} := \max\{U_1, \ldots, U_n\}$.

$$\mathbb{P}[U_{(n)} \le y] = \mathbb{P}[U_1, \ldots, U_n \le y]$$
$$= \prod_{k=1}^{n} \mathbb{P}[U_k \le y] \quad \text{(from independence)}$$
$$= y^n$$

$\Rightarrow$we can generate $U_{(n)}$ by inversion; $U_{(n)}$ is simply the n^{th} root of a $U(0,1)$.

So $U_{(n)} = U_n^{\frac{1}{n}}$ is a r.v. having distribution $\equiv \max\{U_1, \ldots, U_n\}$.

$U_{(1)}, \ldots, U_{(n-1)}$ correspond to the order statistics of a sample of size $n-1$ uniform over $(0, U_{(n)})$...the required result follows by recursion.

(b) Exponential Spacings:

Let $X_1, \ldots, X_{n+1} \overset{iid}{\sim} \text{Exp}(1)$ and set

$$S_k = X_1 + \ldots + X_k, \qquad k = 1, 2, \ldots, n+1.$$

Claim: If we define

$$U_{(k)} = \frac{S_k}{S_{n+1}}, \quad k = 1, \ldots, n$$

$$\Rightarrow U_{(1)} < \ldots < U_{(n)} \quad \text{are ordered} \quad U(0,1)\text{'s.}$$

Proof:

We first consider the joint density of the spacings, then through successive transformation of variables, we show that the joint density of $U(1), \ldots, U(n)$ is as required.

But first...what is the joint distribution of a set of order statistics? In general, given $Y_1, Y_2, \ldots, Y_n \sim f_Y(y)$ then the joint pdf of the corresponding order statistics is

$$f_{Y_{(1)},\ldots,Y_{(n)}}(y_{(1)}, y_{(2)}, \ldots, y_{(n)}) = n! f_Y(y_{(1)}) f_Y(y_{(2)}) \ldots f_Y(y_{(n)})$$

Intuition: there are $n!$ ways of generating $y_1, \ldots, y_n$ that lead to the same ordered sample.

So, the joint pdf of n ordered uniforms is given by

$$f_{U_{(1)},\ldots,U_{(n)}}(u_{(1)}, u_{(2)}, \ldots, u_{(n)}) = n!$$

Now...back to our exponential spacings:

$f_X(x_k) = e^{-x} \Rightarrow$ $$f_X(x_1, \ldots, x_{n+1}) = \prod_{i=1}^{n+1} e^{-x_i} = \exp\left(-\sum_{i=1}^{n+1} x_i\right) \quad x_i \geq 0$$

Using $S_k = \sum_{i=1}^{k} X_i \Rightarrow \begin{cases} X_1 = S_1 \\ X_k = S_k - S_{k-1}, & k = 2, \cdots, n+1 \end{cases}$

we can find the joint density for $S_1, \cdots, S_{k+1}$:

$$f_S(s_1, \cdots, s_{k+1}) = f_X(s_1, s_2 - s_1, \cdots, s_{n+1} - s_n)\,|J|$$

where

$$|J| = \begin{vmatrix} \frac{\partial x_1}{\partial s_1} & \frac{\partial x_2}{\partial s_1} & \cdots & \frac{\partial x_{n+1}}{\partial s_1} \\ \frac{\partial x_1}{\partial s_2} & \frac{\partial x_2}{\partial s_2} & \cdots & \frac{\partial x_{n+1}}{\partial s_2} \\ \vdots & & & \vdots \\ \frac{\partial x_1}{\partial s_{n+1}} & \frac{\partial x_2}{\partial s_{n+1}} & \cdots & \frac{\partial x_{n+1}}{\partial s_{n+1}} \end{vmatrix} \quad \begin{vmatrix} 1 & -1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -1 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{vmatrix} = 1$$

$$\Rightarrow f_S(s_1, s_2, \ldots, s_{n+1}) = \exp\left\{-s_1 - \sum_{k=2}^{n+1}(s_k - s_{k-1})\right\}$$

$$= \exp\{-s_{n+1}\} \quad 0 < s_1 < \cdots < s_{n+1}$$

Now for the second transformation of variables...

$$V_k = \frac{S_k}{S_{n+1}} \quad k = 1, \cdots, n \qquad V_{n+1} = S_{n+1}$$

$$\Rightarrow \begin{cases} S_k = V_{n+1} V_k & k = 1, \cdots, n \\ S_{n+1} = V_{n+1} \end{cases}$$

$$f_V(v_1, \cdots, v_{n+1}) = f_S(v_{n+1}v_1, \cdots, v_{n+1}v_n, v_{n+1})\,|J|$$

$$|J| = \begin{vmatrix} \frac{\partial s_1}{\partial v_1} & \frac{\partial s_2}{\partial v_1} & \cdots & \frac{\partial s_{n+1}}{\partial v_1} \\ \frac{\partial s_1}{\partial v_2} & \frac{\partial s_2}{\partial v_2} & \cdots & \frac{\partial s_{n+1}}{\partial v_2} \\ \vdots & & & \vdots \\ \frac{\partial s_1}{\partial v_{n+1}} & \frac{\partial s_2}{\partial v_{n+1}} & \cdots & \frac{\partial s_{n+1}}{\partial v_{n+1}} \end{vmatrix} = \begin{vmatrix} v_{n+1} & 0 & \cdots & 0 \\ 0 & v_{n+1} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ v_1 & v_2 & \cdots & 1 \end{vmatrix} = v_{n+1}^n$$

$$\Rightarrow f_V(v_1, v_2, \ldots, v_{n+1}) = \exp\{-v_{n+1}\} v_{n+1}^n$$

$$\Rightarrow f_V(v_1, \ldots, v_n) = \int_0^\infty \exp\{-v_{n+1}\} v_{n+1}^n \, dv_{n+1} = I_n$$

Integration by parts yields $I_n = n I_{n-1}$

Recursion $\Rightarrow$ $f_V(v_1, v_2, \ldots, v_n) = n!$ for $v_1 < v_2 < \ldots v_n$,

as required!

6 Markov Chain Monte Carlo Methods

In Chapter 4, we looked at the problem of evaluating expectations with respect to complicated densities. We saw that, using importance sampling, we could avoid sampling directly from the tricky target density, and still achieve unbiased and accurate estimators for the expectation of interest.

Pros of IS: - Independent sample ; estimators are potentially more accurate than eg Crude MC.

Cons of IS: - Requires a prudent choice of auxiliary distribution, which may not always be available or obvious

In this chapter, we consider an alternative approach to the problem. We will see that Markov chain Monte Carlo (MCMC) methods provide an extremely flexible way to sample from many potentially complicated densities, allowing us to perform Monte Carlo estimation without having to resort to the use of an auxiliary target density.

In brief, MCMC methods revolve around the simulation of a class of stochastic processes known as Markov chains. They exploit the fact that, under a set of easily satisfiable regularity conditions, relatively straightforward algorithms exist for *efficiently* generating Markov chains whose elements can (eventually) be considered a sample from a specified distribution. MCMC methods are flexible, computationally attractive, and form an entire branch of computational statistics, in which research is very much active and ongoing!

A Motivating Problem – the Travelling Salesman Problem.

Suppose that a salesman must visit each of n cities, once only, in some order to be determined.

There are $n!$ possible routes, so a direct search for the 'best' route is not feasible.

Let

$$x_i = i\text{th city visited}$$

Let $x = (x_1 \dots x_n)$ be a particular route and $c(x)$ = cost of route x.

eg total distance travelled

Then, we seek: $\arg\min_x c(x)$, where x can take $n!$ possible values

Trick: Define

$$\begin{aligned} p_\lambda(x) &= \frac{\exp(-\lambda c(x))}{\sum_x \exp(-\lambda c(x))} \\ &= \text{const} \times \exp(-\lambda c(x)) \end{aligned}$$

Then $p_\lambda(x)$ is a probability distribution over $\underbrace{1, 2, \ldots, n!}_{\text{each possible route}}$.

Notes:

- If λ is large then:

large $c(x) \Rightarrow$ low probability associated with x
small $c(x) \Rightarrow$ high probability

- As λ increases, only x's which nearly minimise $c(x)$ get any probability.

- $\lambda \to \infty \Rightarrow$ spike at the optimal route, which minimises $c(x)$.

In order to find the most likely x, we could simulate from $p_\lambda(x)$. But how do we simulate a "state" from a discrete probability distribution with a large but finite number of states? If $n!$ is large we can't evaluate the normalizing constant so inversion is not an option.

One way of doing this is to design a Markov Chain whose 'stationary distribution' matches $p_\lambda(x)$ - we can then simulate a realisation of this stochastic process until it converges to this stationary distribution.

6.1 Markov Chains

A stochastic process is simply a collection of indexed RVs: we consider

$$X_t \quad t = 0, 1, 2, \ldots \qquad \text{i.e. discrete time}$$

To start with, we shall suppose that X_t can take any of K possible values (states) that we shall label $\{1, 2, \ldots, K\} = S$ (ie finite, discrete state space)

Definition

A Markov Chain is a discrete time stochastic process

$$\{X_t,\ \ t = 0, 1, 2, \ldots\}$$

with a finite or countable state space, which satisfies the following properties:

(1) Markov Property

$$P(X_n = j \mid X_0 = i_0, X_1 = i_1, \cdots, X_{n-1} = i_{n-1}) = P(X_n = j \mid X_{n-1} = i_{n-1})$$

'Future depends on the present, but not the past'

'Lack of Memory property'

(2) Time Homogeneity

$P(X_{n+1} = j \mid X_n = i)$ is the same for all $n \geq 0$

As a result of these two properties, a Markov chain can be completely characterised by:

- its transition probabilities
- its initial state, ie its distribution

For a discrete state space, the transition probabilities are specified through a transition matrix:

$$P = \begin{pmatrix} p_{00} & p_{01} & \cdots \\ p_{10} & p_{11} & \cdots \\ \vdots & \vdots & \\ & & p_{ij} \\ & & \end{pmatrix}$$

state space in general may be finite or infinite

rows sum to 1

with

$$\begin{aligned} p_{ij} &= P(X_{n+1} = j | X_n = i) \\ &= P(X_1 = j | X_0 = i) \end{aligned}$$

In general, $p_{ij} \neq p_{ji}$

For our purposes, it will be useful to describe the behaviour of the process after a given number of iterations - this is easily done using the n-step transition probabilities

n-step transition probabilities:

$$p_{ij}^{(n)} = P(X_n = j \mid X_0 = i) \qquad \left(p_{ij} = p_{ij}^{(1)}\right)$$

...this is the probability of moving from state i to state j in exactly n steps.

n step transition matrix, $P^{(n)} = \left(p_{ij}^{(n)}\right)$:

$$\begin{aligned} p_{ij}^{(n)} &= P(X_n = j \mid X_0 = i) \\ &= \sum_k P(X_n = j \mid X_1 = k) P(X_1 = k \mid X_0 = i) \\ &= \sum_k p_{kj}^{(n-1)} p_{ik} \quad \Rightarrow \quad P^{(n)} = P P^{(n-1)} \\ &\Rightarrow \quad P^{(n)} = P^n \end{aligned}$$

6.1.1 Chapman-Kolmogorov equations

The most general form of the Chapman-Kolmogorov equations can be applied to any stochastic process, and states that the joint distribution of process realisations at distinct time points can be achieved through marginalisation:

$$\mathrm{P}(X_n = x_n, X_{n+2} = x_{n+2}) = \sum_{x_{n+1}} \mathrm{P}(X_n = x_n, X_{n+1} = x_{n+1}, X_{n+2} = x_{n+2})$$

For Markov processes, considering the corresponding conditional distributions leads to a well-known property regarding the n-step transition probabilities; this is the most commonly referenced form of the Chapman-Kolmogorov equations.

Markov property

$$P(X_n = n)\, P(X_{n+2} = x_{n+2} \mid X_n = x_n) = \sum_{x_{n+1}} P(X_{n+2} = x_{n+2} \mid X_{n+1} = x_{n+1}) P(X_{n+1} = x_{n+1} \mid X_n = x_n) \cdot P(X_n = n)$$

$$\vdots$$

$$P(X_{n+m} = j \mid X_0 = i) = \sum_k P(X_{n+m} = j \mid X_n = k) P(X_n = k \mid X_0 = i)$$

$$p_{ij}^{(n+m)} = \sum_k p_{ik}^{(m)} p_{kj}^{(n)}$$

6.1.2 Initial distribution

We now specify notation for the distribution of the initial state of the Markov chain, and we show that this can be used in conjunction with the transition probabilities to fully specify the distribution of the chain at each time point.

Let $\pi_i^{(0)} = \mathrm{P}(X_0 = i)$ and use $\underline{\pi}^{(0)}$ to denote the row vector, i.e. the initial distribution over the discrete state space.

What is the distribution at iteration n, $\underline{\pi}^{(n)}$?

$$\pi_i^{(n)} = P(X_n = i) = \sum_j P(X_n = i, X_{n-1} = j)$$
$$= \sum_j P(X_n = i \mid X_{n-1} = j) P(X_{n-1} = j)$$
$$= \sum_j \pi_j^{(n-1)} p_{ji}$$
$$\Rightarrow \underline{\pi}^{(n)} = \underline{\pi}^{(n-1)} P$$
$$\Rightarrow \underline{\pi}^{(n)} = \underline{\pi}^{(0)} P^n$$

Thus, $\underline{\pi}^{(n)}$ is fully specified by P and $\underline{\pi}^{(0)}$.

6.1.3 Stationary Distributions

We will consider a subclass of Markov chains: those whose distribution is invariant under multiplication by the transition matrix P.

$$\underline{\pi}^{(n)} = \underline{\pi}^{(n-1)} = \underline{\pi}; \quad \underline{\pi} = \underline{\pi} P$$

ie. we want the elements of the chain to be identically distributed

Markov chains with this property are said to have a stationary distribution, $\underline{\pi}$, defined as follows:

Definition

$\underline{\pi}$ is a stationary distribution iff

(i) $\pi_i \geq 0 \quad \forall i$.

(ii) $\sum_i \pi_i = 1$.

(iii) $\pi_j = \sum_i \pi_i p_{ij} \quad \forall j \ (\underline{\pi} = \underline{\pi} P)$.

Properties of stationary distributions

(i) $\underline{\pi}^{(0)} = \underline{\pi} \Longrightarrow \underline{\pi}^{(n)} = \underline{\pi} \quad \forall n$

$\underline{\pi}^{(m)} = \underline{\pi} \Rightarrow \underline{\pi}^{(n)} = \underline{\pi} \quad \forall n \geq m$

"process is in equilibrium"

(ii) Suppose state space is finite, and suppose $\underline{\pi}^{(n)}$ converges as $n \to \infty$. Then the limit must be a stationary distribution.

Proof:

Suppose $\pi_i^{(n)} \longrightarrow \pi_{*i}$

Since $\pi_i^{(n)} = \sum_j \pi_j^{(n-1)} p_{ji}$

As $n \to \infty$, the same result holds (limit of a finite sum is the sum of the limits).

$\pi_i = \sum_j \pi_j p_{ji}$

Finding the stationary distribution

Supposing a stationary distribution exists, it can be found by solving the equations

$$\underline{\pi} = \underline{\pi} P \qquad \sum \pi_i = 1,$$

So if $\underline{\pi} = (\pi_1, \ldots, \pi_d)$ we have $d+1$ equations for d unknowns $\Rightarrow$ one equation will be redundant.

For some P, $\exists$ a unique stationary distribution...

...for some P, $\exists$ more than one stationary distribution...

...and for some P, $\exists$ no stationary distribution. (for when the state space is infinite)

It will be useful to us to establish the conditions under which there exists a stationary distribution.

Furthermore, we note that the existence of a stationary distribution does not, in general, guarantee that the chain's convergence to that distribution.

We will also, therefore, establish the conditions under which a limiting distribution exists for the chain. This will allow us to build Markov chains that converge to a stationary distribution that we specify.

Definition

If a Markov chain converges to its stationary distribution, then this is also referred to as its equilibrium distribution.

6.1.4 Reversibility

We will focus on a (further) subclass of Markov chains - those with the property of reversibility. We will see that this is a sufficient condition to guarantee the existence of a stationary distribution.

Recall our definition of a stationary distribution: we require

$$\pi_i = \sum_j \pi_j p_{ji}$$

It is straightforward to see that this will follow if we make the following, simpler restriction on P:

$$\pi_j p_{ji} = \pi_i p_{ij}$$

summing over i $\Rightarrow$ $\pi_j \sum_i p_{ji} = \sum_i \pi_i p_{ij}$ ($\sum_i p_{ji} = 1$)

This is known as the **detailed balance equation**, and a Markov chain that satisfies this property will have a stationary distribution by design.

We now consider the properties of periodicity and reducibility for Markov chains in general, and show that these are sufficient to guarantee convergence of a Markov chain to its invariant distribution $\underline{\pi}$ (assuming its existence).

6.1.5 Irreducibility

Definition- Classification of States

For a given Markov chain, state j communicates with state i ($i \leftrightarrow j$) if

$$p_{ij}^{(n)} > 0 \quad \text{and} \quad p_{ji}^{(m)} > 0 \quad \text{for some} \quad n, m \geq 0.$$

ie each state communicates with itself, by convention.

Result: $\leftrightarrow$ is an equivalence relationship, since:

(1) $i \leftrightarrow i$ (reflexive)

(2) $i \leftrightarrow j \Leftrightarrow j \leftrightarrow i$ (symmetric)

(3) $i \leftrightarrow j, j \leftrightarrow k \Rightarrow i \leftrightarrow k$ (transitivity) (Chapman-Kolmogorov).

and so the state space can be divided into disjoint equivalence classes s.t.

$$i \leftrightarrow j \Leftrightarrow i, j \quad \text{are in the same class;}$$

we call these classes of state.

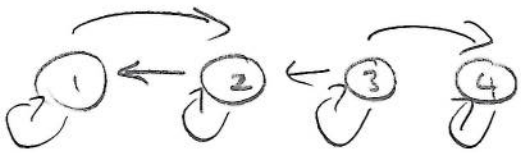
Note:

1. $p_{ij} = 0 \ \forall \ j \neq i \Leftrightarrow i$ is an absorbing state.
2. $p_{ij} = 0 \ \forall \ i \in C, \ \forall \ j \notin C \Leftrightarrow C$ is a closed class.

To determine classes, need only look at the structure of zero and non-zero elements of P.

Example:

$$P = \begin{pmatrix} + & + & 0 & 0 \\ + & + & 0 & 0 \\ 0 & + & + & + \\ 0 & 0 & 0 & + \end{pmatrix}$$



"+" represents a probability > 0

Classes: $\{1, 2\}$; $\{3\}$; $\{4\}$

closed open absorbing state.

Definition- Irreducible Chains

If the state space of a given Markov chain consists of a single class (necessarily closed) then the chain is said to be irreducible.

⇒ a stationary distribution

Note - Finite Irreducible Chains

If a Markov chain is defined on a finite state space and is irreducible, then there exists a unique stationary distribution satisfying

$$\underline{\pi} = \underline{\pi} P \qquad \text{and} \qquad \sum \pi_i = 1.$$

Furthermore,

$$\pi_i > 0 \qquad \forall i$$

ie for finite state spaces, we do not need the chain to be reversible - if it is irreducible, a unique stationary distribution is already guaranteed.

6.1.6 Periodicity

The period of state i is the greatest common divisor of

$$\{n : p_{ii}^{(n)} > 0\}$$

← minimum number of steps to return to i

Example

infinite state space

1. Unbounded simple random walk:

$X_{n+1} = X_n + 1$ with prob p
$X_{n+1} = X_n - 1$ with prob $1-p$

$$\{n : p_{ii}^{(n)} > 0\} = \{2, 4, 6, \ldots\} \quad \forall\ i$$

each state will have period 2

2.

$$P = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}$$

$\{n : p_{11}^{(n)} > 0\} = \{2, 3, 4, \ldots\} \Rightarrow$ state 1 has period 1

$\{n : p_{22}^{(n)} > 0\} = \{1, 2, 3, \ldots\} \Rightarrow$ state 2 "

$\{n : p_{33}^{(n)} > 0\} = \{2, 3, 4, \ldots\} \Rightarrow$ state 3 "

Definition - Aperiodicity

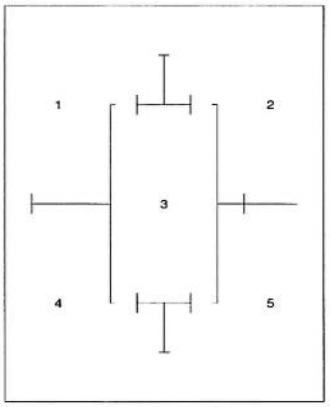
- A state is said to be **aperiodic** if its period is 1. ($p_{ii} > 0 \Rightarrow$ state i is aperiodic)
- A Markov chain $\{X_t\}$ is **aperiodic** if all states in the corresponding state space are aperiodic.

Theorem 6.1

If $i \leftrightarrow j$ then i and j have the same period. Thus, all states of an equivalence class have the same period.

$\Rightarrow$ If $\{X_t\}$ is irreducible, all of its states will have the same periodicity

Another example... Mouse in maze (see Exercises 6)...



In any room, mouse selects at random any of the possible doors, each equally likely.

$$P = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \end{pmatrix}$$

Key property : MC is irreducible

State 1 : $\{n : p_{11}^{(n)} > 0\} = \{2, 3, 4, \dots\}$

$\Rightarrow$ period is 1

$\Rightarrow$ state 1 is aperiodic

& irreducibility $\Rightarrow$ MC is aperiodic

Theorem 6.2 (MARKOV CHAIN CONVERGENCE)

Suppose $\{X_n\}$ is an aperiodic, irreducible Markov chain with finite state space and unique stationary distribution π, then

$$P(X_n = i) \longrightarrow \pi_i \quad \text{as } n \to \infty \qquad \forall i \in S$$

NOTE: If the chain is irreducible, we need only show aperiodicity for a single state.

6.2 Sampling from the stationary distribution

Suppose we want to obtain samples from some distribution $\underline{\pi}$, which has a large number of states and so cannot be sampled from directly.

Idea: Set up an irreducible, aperiodic MC (i.e. define P), with stationary distribution $\underline{\pi}$. Then run this chain until it settles down to $\underline{\pi}$ – states will then be generated with the correct probabilities.

We have considered the problem of finding $\underline{\pi}$ given a specified P - now, we must find P given a specified $\underline{\pi}$ as its stationary distribution. Can we do this, and at the same time, ensure that the resulting chain is convergent?

6.2.1 Metropolis Algorithm

We can construct the required P by considering a probabilistic transition mechanism. We choose any symmetric transition matrix Q, with elements q_{ij}. Then, we define our transition mechanism as follows:

such that $\sum_j q_{ij} = 1$ & $0 \leq q_{ij} \leq 1$

Suppose the chain is in state i...

- we select state j as a candidate for the next state of the chain, with probability q_{ij},
- and we then move to state j with probability

$$\min\left\{1, \frac{\pi_j}{\pi_i}\right\},$$

otherwise, we stay at state i.

Note: if $\pi_j \geq \pi_i$, we'll definitely move.

This defines P in the following way:

$$p_{ij} = q_{ij} \min\left\{1, \frac{\pi_j}{\pi_i}\right\} \qquad i \neq j$$

$$p_{ii} = q_{ii} + \sum_{j \neq i} q_{ij} \max\left\{0, 1 - \frac{\pi_j}{\pi_i}\right\}$$

q_{ii}: prob of choosing candidate i

$\max\{0, 1 - \frac{\pi_j}{\pi_i}\}$: prob of rejecting the proposed move to $j \neq i$

P defined in this way is certainly a transition matrix (as Q is), but will running the chain for a long time give us $\underline{\pi}$ as the limiting distribution?

- For this, we need to show that P is irreducible, aperiodic and that $\underline{\pi} = \underline{\pi}P$.

For the P designed above, we have

$$\begin{aligned}
\pi_i p_{ij} &= \pi_i \min\left\{1, \frac{\pi_j}{\pi_i}\right\} q_{ij} \\
&= \min\left\{\pi_i, \pi_j\right\} q_{ij} \\
&= \min\left\{\pi_j, \pi_i\right\} q_{ji} \\
&= \pi_j \min\left\{1, \frac{\pi_i}{\pi_j}\right\} q_{ji} \\
&= \pi_j p_{ji}
\end{aligned}$$

...i.e. P satisfies the detailed balance equations (by design), and so the generated Markov chain will be reversible and have $\underline{\pi}$ as a stationary distribution.

If we can now show that the chain is irreducible and aperiodic, then we guarantee that $\underline{\pi}$ will be the unique stationary distribution, and furthermore that the chain will be convergent.

Irreducibility If Q is irreducible, then so is P ($p_{ij} > 0$ iff $q_{ij} > 0$), so choose an irreducible Q!

Aperiodicity If we can show $p_{ii} > 0$, then state i is aperiodic, and thus all states are aperiodic (as there is only one class).

$$p_{ii} = q_{ii} + \sum_{j \neq i} \max\left\{0, 1 - \frac{\pi_j}{\pi_i}\right\} q_{ij}$$

i.e. $p_{ii} \geq q_{ii}$ – so we should choose Q with $q_{ii} > 0$, for some $i \in S$.

In short, choose Q that guarantees aperiodicity & irreducibility

NB: only need to know $\underline{\pi}$ up to some normalizing constant, as it only enters the algorithm through $\frac{\pi_j}{\pi_i}$

Notes:

- Other prescriptions include that of Barker (1965) who replaced

$$\min\left\{1, \frac{\pi_j}{\pi_i}\right\} \quad \text{with} \quad \frac{\pi_j}{\pi_i + \pi_j}$$

This defines P as

$$\begin{aligned} p_{ij} &= q_{ij}\frac{\pi_j}{\pi_i + \pi_j} \\ p_{ii} &= \frac{q_{ii}}{2} + \sum_{j \neq i} q_{ij}\frac{\pi_i}{\pi_i + \pi_j} \end{aligned}$$

- Hastings (1970, Biometrika) gives a general class of algorithms that include both of the above.

6.2.2 Metropolis-Hastings for discrete state spaces

In a similar way to the Metropolis algorithm, the Metropolis-Hastings procedure generates a Markov chain with specified equilibrium distribution by iteratively proposing candidate states and accepting or rejecting these proposals with a calculated probability.

Metropolis-Hastings Procedure (discrete state space):

Suppose that the chain is in state i...

- select state j as a candidate state, according to some predefined transition probability q_{ij};
- then accept this candidate as the next state of the chain with probability

$$\alpha_{ij} = \min\left\{1, \frac{\pi_j q_{ji}}{\pi_i q_{ij}}\right\}.$$

Think of this as $\frac{\pi_j}{q_{ij}} / \frac{\pi_i}{q_{ji}}$

The Markov chain generated by the discrete version of the Metropolis-Hastings sampler will have a transition matrix P specified by

$$p_{ij} = \min\left\{1, \frac{\pi_j q_{ji}}{\pi_i q_{ij}}\right\} q_{ij} \qquad i \neq j$$

$$p_{ii} = q_{ii} + \sum_{j \neq i} \max\left\{0, 1 - \frac{\pi_j q_{ji}}{\pi_i q_{ij}}\right\} q_{ij}$$

and, as before, we have that this satisfies the detailed balance conditions by design:

Suppose $\pi_j q_{ji} \geq \pi_i q_{ij}$

$\Rightarrow p_{ij} = q_{ij}$, $p_{ji} = \frac{\pi_i q_{ij}}{\pi_j}$

$\Rightarrow \pi_i p_{ij} = \pi_j p_{ji}$

... and swapping the indices gives the required result for $\pi_j q_{ji} \leq \pi_i q_{ij}$

So the two-step update procedure above will specify the required transition properties in the resulting Markov chain - do we need to specify anything else?

We have to initialise somehow! But this is easy - the resulting Markov chain will be irreducible (by design) and so we can start from anywhere in our state space, and we will eventually converge.

Note that, in order to recover the original Metropolis algorithm, all that is required is to choose a symmetric matrix of proposal probabilities $Q = \{q_{ij}\}$. More importantly, however...we are no longer restricted to symmetric proposal mechanisms!

In particular, with the Metropolis-Hastings sampler, we are allowed to propose candidate values for our chain independently of the chain's current state

- this is Independent Metropolis-Hastings, sometimes useful for Bayesian inference.

6.2.3 Metropolis-Hastings for continuous state spaces

We have developed the Metropolis-Hastings algorithm for exploring discrete state spaces - this has allowed us to rigorously explore the conditions required for the resulting Markov chain to converge to a specified stationary distribution.

The MH procedure for sampling from a continuous target distribution is analagous to the discrete case, though rigorous proofs for the convergence of the resulting chain are more complicated; we satisfy ourselves with extending to the continuous case by simple comparison with the discrete case.

Suppose we wish to sample from the target density $f(x)$, defined on a continuous state space $\mathcal{X}$. As in the discrete MH procedure, we must specify a proposal

mechanism for each step of the procedure. Here, we require a *proposal density* $q(y|x) \geq 0$, which satisfies

$$\int_{\mathcal{X}} q(y|x)dy = 1.$$

We must also initialise our procedure - this is once again a relatively straightforward affair. As long as our transition density $q(y|x)$ allows us to move from any part of the state space to any other part of the state space, in a finite number of iterations, then the Markov chain will be irreducible and thus converge to the required target from any arbitrary starting point chosen from within $\mathcal{X}$. Note that we can also choose to sample from a specified arbitrary initial distribution $\pi^{(0)}$.

Metropolis-Hastings Procedure (continuous state space):

1. Initialise the chain: start from an arbitrary X_0, possibly sampled from $\pi^{(0)}$. Set $n = 1$

2. Given $X^{(n-1)} = x$, generate a candidate value $Y = y$ from the proposal density $q(y|x)$.

3. Set $X^{(n)} = y$ with probability $\alpha(x, y)$, where

$$\alpha(x,y) = \min\left\{1, \frac{f(y)\, q(x|y)}{f(x)\, q(y|x)}\right\}$$

   otherwise, set $X^{(n)} = x$.

4. Replace n by $n + 1$ and return to Step 2.

Burn-in Periods In both the discrete and continuous cases, the MH procedure is guaranteed to converge - before convergence occurs, however, the Markov chain cannot be used to represent a sample from the target distribution.

It is therefore common to allow a period of *burn-in* at the start of the algorithm, during which convergence is assumed to have not yet occurred.

After a specified number of samples, the initial part of the chain is typically discarded, and the rest of the chain may be treated as a random sample from the target distribution of interest.

For example, a typical use of the algorithm might treat the first $B = 10,000$ iterations as a burn-in sample, then continue for another 100,000 iterations to provide what is then treated as a random sample from the desired distribution.

The chosen length of burn-in period B is often subjective, and affected as much by available computer time as by formal convergence considerations, though convergence diagnostics may also be used.

Dependence After discarding the burn-in sample, we are left with a statistically dependent sample from the target distribution. This could be slightly problematic... as MC integration requires iid samples!

Thankfully, we need not worry (much). It has been shown by Meyn and Tweedie (*Markov Chains and Stochastic Stability*, CUP, 2009) that under the existing assumptions of irreducibility and aperiodicity, the Monte Carlo estimate resulting from an M-length Markov chain realisation $X_{B+1}, \ldots, X_{B+M}$ will satisfy the Strong Law of Large Numbers (SLLN) for any test function ϕ such that $\int_{\mathcal{X}} |\phi(x)| f(x) dx < \infty$:

$$\lim_{M \to \infty} \frac{1}{M} \sum_{i=1}^{M} \phi(X_{B+i}) = \mathbb{E}[\phi(X)] \quad \text{with probability 1.}$$

For moderate- length post-burn-in samples resulting from the MH procedure, a common step to take is to further discard all but every k-th iteration, where k is some sensible lag, chosen subjectively.

- this is called thinning
- it can be very wasteful!

6.2.4 Scaling

The key quantity that has to be specified in the Metropolis-Hastings algorithm is the proposal density $q(y|x)$. This is often simplified by writing

$$q(y|x) = \frac{1}{h} g\left(\frac{y-x}{h}\right)$$

for some density g symmetric about 0 and a *scaling constant* h; note that in this case we automatically have $q(y|x) = q(x|y)$.

We may take g to be some standard form such as normal or uniform; the critical issue then becomes how to choose h so that the algorithm converges in reasonable time:

- If h is too large, then the proposal mechanism will explore the state space well, but will often propose candidate values in regions where the resulting ratio of target densities is low, causing high probability of rejection.
- In contrast, if h is small, then we may get a low rejection probability at each step, but we have poor exploration of the state space and high probability in the resulting chain.

In a remarkable result, Roberts, Gelman and Gilks (1997) argued that the correct scaling constant is one that leads to an overall acceptance rate (average value of α) of about 0.23. The mathematical derivation of this result involves a number of steps that take us rather far from the original Hastings-Metropolis algorithm, for example, assuming the dimension of the sampling space tends to ∞...nevertheless, it is found in practice that this rule gives good guidance to the optimal scaling (...even in cases not formally covered by their theorem.)

It can be hard in practice to find h to achieve some predetermined acceptance rate, so Gilks *et al.* (1995) recommend, as a rule of thumb, trying to achieve an acceptance rate between 15% and 50% ...this seems good enough for most practical purposes.

6.3 Metropolis-Hastings for Bayesian inference

Suppose we adopt a Bayesian perspective. Our interest is now in establishing information about the distribution of a parameter $\theta \in \Theta \subseteq \mathbb{R}^d$, given some prior belief about the parameter's distribution, represented through the prior density $p(\theta)$, and given observation on some data dependent on the parameter with likelihood $\ell(x|\theta)$.

We therefore seek to perform inference with respect to the posterior density

$$\pi(\theta|x) \propto \ell(x|\theta)\, p(\theta)$$

and it is common to use simulation-based inference methods, especially when

- the state space of Θ is high-dimensional, ie $d \gg 1$
- the normalizing constant is analytically intractable.

We turn to Monte Carlo integration...in particular, to the use of MCMC methods!

A Metropolis-Hastings procedure may be designed for simulating a Markov chain that can be considered a sample from $\pi(\theta|x)$:

(Continuous) MH Procedure for Bayesian Inference:

1. Initialise the chain : sample $\theta^{(0)} \sim p(\theta)$. Set $n = 1$
2. Given $\theta^{(n-1)} = \theta$, generate a candidate value η from a chosen proposal density $q(\eta|\theta)$.
3. Set $\theta^{(n)} = \eta$ with probability $\alpha(\theta, \eta)$, where
$$\alpha(\theta, \eta) = \min\left\{ \frac{\pi(\eta|x)}{\pi(\theta|x)} \frac{q(\theta|\eta)}{q(\eta|\theta)}, 1 \right\}$$
otherwise, set $\theta^{(n)} = \theta$.
4. Replace n by $n+1$ and return to Step 2.

Now consider the ratio of densities in the acceptance probability:

$$\frac{\pi(\eta|x)q(\theta|\eta)}{\pi(\theta|x)q(\eta|\theta)} = \frac{\ell(x|\eta)p(\eta)}{\ell(x|\theta)p(\theta)}\frac{q(\theta|\eta)}{q(\eta|\theta)}$$

We can make two important notes:

- We only need the target posterior up to proportionality
- We can simplify the acceptance ratio using a particular choice of q:

Independent MH $\longrightarrow$ $q(\eta|\theta) = p(\eta) \Rightarrow \alpha(\theta,\eta) = \min\left\{\frac{\ell(x|\eta)}{\ell(x|\theta)}, 1\right\}$

When the state-space of the target distribution is high-dimensional, Metropolis-Hastings provides a flexible approach to performing simulation based inference...but there is a more efficient method...

6.4 Gibbs sampling for continuous state spaces

Suppose we have a d-dimensional target density, $f(x)$, $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$. The Gibbs sampler provides a relatively efficient method for generating multidimensional Markov chains, given knowledge of the conditional dependence structure of the target distribution.

In order to employ the Gibbs sampler, we divide the variable into a number of components of smaller dimension. We will split the variable into single-dimensional components (for notation's sake) though this is not strictly necessary.

The Gibbs sampler proceeds by separately updating each component of the variable, conditional on the most up-to-date version of the remaining components. A requirement for implementing the Gibbs sampler is therefore the knowledge of the *full conditionals*:

$$f(x_j|x_{-j}) = \frac{f(x)}{\int_{\mathbb{R}} f(x)\,dx_j}$$

$$x_{-j} = (x_1, \cdots, x_{j-1}, x_{j+1}, \cdots, x_d) \qquad j = 1, \cdots, d$$

Gibbs Sampler:

1. Initialise the multivariate chain at an arbitrary initial vector, say $X^{(0)} = (X_1^{(0)}, \ldots, X_d^{(0)})$. Set $n = 1$

2. Given $X^{(n-1)}$, sample each component of $X^{(n)}$ from the full-conditionals:

   - sample $X_1^{(n)} \sim f(x_1 | X_2^{(n-1)}, X_3^{(n-1)}, \ldots, X_d^{(n-1)})$
   - sample $X_2^{(n)} \sim f(x_2 | X_1^{(n)}, X_3^{(n-1)}, \ldots, X_d^{(n-1)})$
   - $\vdots$
   - sample $X_d^{(n)} \sim f(x_d | X_1^{(n)}, X_2^{(n)}, \ldots, X_d^{(n-1)})$

3. Replace n by $n + 1$ and return to Step 2.

The idea here is that the full conditional densities are of (perhaps significantly) lower dimension than the joint target density of interest.

- If the full conditionals are one-dimensional, we can probably use one of the many procedures from Ch 3-5!
- If they are low-dimensional yet also difficult to sample from, ... we could use Metropolis-within-Gibbs (actually Metropolis-Hastings-within-Gibbs).

The above algorithm is 'deterministic scan' Gibbs... we could also have the more general 'random scan' Gibbs Sampler, where, at each iteration, we cycle through our full-conditionals in a random order.
- We must ensure that each component is updated exactly once at each iteration of the sampler.

6 Monte Carlo Tests

Suppose we have a null hypothesis H_0, represented by a completely specified model, and a test statistic T, such as a goodness-of-fit statistic, for which small values indicate departure from H_0. Denote the observed data by y and let the observed value of T be $t = T(y)$.

To perform a pure significance test we need to know the distribution of T under H_0. This may be difficult or impossible to obtain analytically, but it may be possible to simulate from the model to produce m samples $y_1^*, \ldots, y_m^*$ and the corresponding values $t_1^*, \ldots, t_m^*$ of T. We could then estimate the distribution of T under H_0 by the empirical distribution of $(t_1^*, \ldots, t_m^*)$, or a smoothed version thereof. Informally, we estimate the critical point of a level α test by the 100αth percentile of $(t_1^*, \ldots, t_m^*)$.

Monte Carlo tests (Barnard, 1963) are a related idea. If H_0 is true, we have $m+1$ values from the distribution of T, m by simulation and one by observation. Thus the probability that t is the kth smallest or smaller of $\{t, t_1^*, \ldots, t_m^*\}$ is $k/(m+1)$, ignoring ties. [Assume a continuous distribution for T.] If we choose k and m so that $\alpha = k/(m+1)$ is a conventional significance level (e.g. 0.01 or 0.05), we have a 'Monte Carlo test', which rejects H_0 if t is the kth smallest or smaller of the $m+1$ values. [Two sided versions of the Monte Carlo test follow an obvious construction].

The Monte Carlo test has a random critical point and so 'blurs' the critical region, though the test is 'exact':

$$P(\text{Type I error}) = P(\text{reject } H_0 \mid H_0 \text{ true}) = \frac{k}{m+1} = \alpha$$

Let F be the distribution function of T under H_0. Recall that

$$u = F(T) \sim U(0,1)$$

A 'conventional test of size α', possible if F were known, rejects H_0 if

$$u = F(t) = P(T \leq t \mid H_0) < \alpha. \tag{6.1}$$

The Monte Carlo test rejects H_0 if

$$t < t_{(k)}, \tag{6.2}$$

where $t_{(k)}$ is the kth order statistic of $(t_1^*, \ldots, t_m^*)$. So, the Monte Carlo test rejects H_0 with probability p, where

$$p = P\left(r \text{ simulated values of } T \text{ are} \le t,\ 0 \le r \le k-1\right).$$
$$= \sum_{r=0}^{k-1} P\left(r \text{ simulated values of } T \le t\right)$$
$$= \sum_{r=0}^{k-1} \binom{m}{r} u^r (1-u)^{m-r}, \quad u = F(t).$$

This should be interpreted as the proportion of times the Monte Carlo test will reject H_0 with $t = F^{-1}(u)$ as observation. Note: we are not assuming $t \sim F_T$, just using F^{-1} as a transformation as we have $t_i^* \sim F_T$.

Marriott (1979) tabulates p, e.g. for $\alpha = 0.05$ we have

$F_T(t)$ ← u

	p	0.1	0.075	0.05	0.025	0.01
	1	0.135	0.227	0.377	0.618	0.826
	2	0.088	0.199	0.413	0.745	0.942
k	3	0.057	0.171	0.429	0.817	0.979
	4	0.038	0.148	0.438	0.864	0.992
	5	0.025	0.128	0.445	0.897	0.997
	10	0.004	0.065	0.461	0.971	–

For small k, the effect of blurring can be substantial. However, the conventional test (6.1) and the Monte Carlo test (6.2) will only give different decisions a significant proportion of the time when t corresponds to a p-value near α.

Compare the power function $\beta^{(m)}(\alpha)$ of the Monte Carlo test with the power $\beta(\alpha)$ of the conventional test. We expect, and it is true, that $\beta^{(m)} \le \beta$, but how large can the power loss be?

Power = $1 - P(\text{Type II error}) = P(\text{reject } H_0 \mid H_0 \text{ false})$.

The conventional test (6.1) rejects H_0 if $T < F^{-1}(\alpha)$ and has power against an alternative denoted by F_θ of

$$\beta(\alpha) = P\left(T < F^{-1}(\alpha) \mid T \sim F_\theta\right) = F_\theta\left(F^{-1}(\alpha)\right).$$

Similarly, the power of the Monte Carlo test (6.2) against this alternative is

$$\begin{aligned}
\beta^{(m)}(\alpha) &= P(T < T_{(k)} \mid T \sim F_\theta, T_1^*, \ldots, T_m^* \sim F) \\
&= \int_{-\infty}^{\infty} \sum_{r=0}^{k-1} \binom{m}{r} F(t)^r [1 - F(t)]^{m-r} dF_\theta(t) \\
&= \int_{-\infty}^{\infty} \int_{F(t)}^{1} b(\alpha, m, \xi) d\xi dF_\theta(t) \\
&= \int_0^1 F_\theta\big(F^{-1}(\xi)\big) b(\alpha, m, \xi) d\xi \\
&= \int_0^1 \beta(\xi) b(\alpha, m, \xi) d\xi, \qquad (6.3)
\end{aligned}$$

where $b(\alpha, m, \cdot)$ is the pdf of a beta distribution with parameters $\alpha(m+1)$ and $(1-\alpha)(m+1)$.

Jöckel (1986) uses (6.3) to establish the following results:

Theorem 7.1

Suppose $\beta(\cdot)$ is concave on $[0, 1]$. Then

$$\beta^{(m)}(\alpha) \uparrow \beta(\alpha) \text{ as } m \to \infty.$$

So if m is big, the Monte Carlo test is OK. The amount we lose by using a Monte Carlo test can be bounded.

Theorem 7.2

Suppose $\beta(\cdot)$ is concave on $[0, 1]$ with $\beta(0) = 0$ and $\beta(1) = 1$. Then

$$\frac{\beta^{(m)}(\alpha)}{\beta(\alpha)} \geq 1 - \frac{E|Z - \alpha|}{2\alpha}$$

where

$$Z \sim \text{Beta}\big(\alpha(m+1), (1-\alpha)(m+1)\big).$$

Relative power of the Monte Carlo test.

For $\alpha = 0.05$, the approximate bounds on $\beta^{(m)}(\alpha)/\beta(\alpha)$ are:

$$\begin{aligned} 0.64 \quad &\text{at} \quad m = 19, \\ 0.83 \quad &\text{at} \quad m = 99, \\ \text{and } 0.95 \quad &\text{at} \quad m = 999. \end{aligned}$$

All in all, study of properties of Monte Carlo tests suggests avoiding small values of k, which blur the critical region a lot and lose power. Since $\alpha = k/(m+1)$, this implies m is large. Usually $m = 99$ will be sufficient.

Example

Consider n random points in $[0,1]\times[0,1]$. The null hypothesis is one of uniformity, to be tested against one of inhibition between points. Let $T = \#$ 'close' pairs of points. Small values of T reject H_0. A Monte Carlo test of the significance of t is performed by generating m samples of n points uniformly on $[0,1]\times[0,1]$. Notice that in performing the test we can stop when k samples t_i^* are smaller than t, for then we know we will not reject. Also, if $m-k+1$ values exceed t, we will certainly reject and can stop. This will be useful if the fit is good, so t is typical null value, but will not help if the true p-value for t is small.

Remarks

1. Often the situation described, where the null hypothesis distribution of T does not involve any nuisance parameters, is induced by conditioning on the sufficient statistic under H_0 for the nuisance parameters.

2. The values $t, t_1^*, \ldots, t_m^*$ do not need to be independent outcomes: the method remains valid as long as they are exchangeable outcomes, meaning that the joint density of $T, T_1^*, \ldots, T_m^*$ under H_0 is invariant under permutation of its arguments. This allows application of Monte Carlo tests in complicated problems, by generalized Monte Carlo tests.

8 Resampling Techniques

8.1 The Jackknife

Consider the classical method of estimating μ, the mean of a population, using data $X_1, \ldots, X_n$.

Estimate μ by

$$\hat{\mu} = \bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

To get some idea of the performance of the estimator $\bar{X}$, we can of course estimate its variance using the sample variance:

$$\mathrm{Var}\left[\bar{X}\right] = \frac{1}{n}\mathrm{Var}\left[X\right] \implies \widehat{\mathrm{Var}}\left[\bar{X}\right] = \frac{S^2}{n} \qquad (*)$$

an estimator for the estimator $\bar{X}$'s variance!

where

$$S^2 = \widehat{\sigma^2} = \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2$$

Note: S^2 is unbiased for σ^2.

Hence, we obtain the usual estimator of the variance of $\bar{X}$:

$$\widehat{\mathrm{Var}}(\bar{X}) = \frac{S^2}{n} \qquad (*)$$

Consider an alternative route:

- Construct n distinct $(n-1)$-length samples from our original sample $X_1, \ldots, X_n$; do this by leaving out each datapoint in turn.

$$\underline{X}_{-i} = \underline{X}_{(i)} = (X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n), i = 1, \ldots, n$$

- Use each sample $\underline{X}_{(i)}$ to obtain a different estimate of μ:

$$\bar{X}_{(i)} = \frac{1}{n-1}\sum_{j=1}^{n-1} X_{(i),j}$$

Now, we can recover our original estimator $\bar{X}$ by simply taking the mean of our n sample means:

$$\frac{1}{n}\sum_{i=1}^{n}\bar{X}_{(i)} = \frac{1}{n}\frac{1}{n-1}\sum_{i=1}^{n}\sum_{j=1}^{n-1} X_{(i),j} = \frac{1}{n(n-1)}\sum_{i=1}^{n}(n-1)X_i = \bar{X}$$

and, moreover, we can estimate the variance of this estimator using a corrected sample variance of the n sample means:

$$\widehat{\mathrm{Var}}\left[\bar{X}\right] = \frac{n-1}{n}\sum_{i=1}^{n}\left(\bar{X}_{(i)} - \bar{X}\right)^2, \qquad (**)$$

where the correction is chosen to ensure that $\widehat{\mathrm{Var}}\left[\bar{X}\right] = S^2/n$, i.e. $(**) = (*)$.

To see this, sub. in $\bar{X}_{(i)} = \frac{1}{n-1}\left(n\bar{X} - X_i\right)$

Now, in the general case, where there is no handy way of expressing the estimator variance in terms of the sample variance (as with $(*)$), the above gives an alternative route.

In general... suppose we have a statistic θ, with an estimator

$$\hat{\theta} = \hat{\theta}\left(X_1, \ldots, X_n\right),$$

we perform 'Jackknife resampling' to obtain

$$\hat{\theta}_{(i)} = \hat{\theta}\left(\underline{X}_{(i)}\right) = \hat{\theta}\left(X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n\right), \qquad i = 1, \ldots, n.$$

This allows us to calculate the **Jackknife variance estimator** for $\hat{\theta}$:

$$\widehat{\mathrm{V}}_{jack}\left[\hat{\theta}\right] = \frac{n-1}{n}\sum_{i=1}^{n}\left(\hat{\theta}_{(i)} - \hat{\theta}_{(\cdot)}\right)^2,$$

where

$$\hat{\theta}_{(\cdot)} = \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_{(i)}.$$

Furthermore, in cases where $\hat{\theta}$ is a biased estimator for θ, we can also estimate the bias using the **Jackknife bias estimator** for $\hat{\theta}$:

$$\widehat{\mathrm{bias}}_{jack}\left[\hat{\theta}\right] = (n-1)\left(\hat{\theta}_{(\cdot)} - \hat{\theta}\right)$$

Note: this is biased!

We can therefore obtain a reduced-bias estimator; the Jackknife bias-reduced estimator of Θ:

$$\tilde{\Theta}_{jack} = \hat{\Theta} - (n-1)\left(\hat{\Theta}_{(\cdot)} - \hat{\Theta}\right)$$
$$= n\hat{\Theta} - (n-1)\hat{\Theta}_{(\cdot)}$$

8.2 The Delete-d Jackknife

Note that the above can be generalised to the case where we create subsamples of the data by deleting d datapoints instead of 1. The size of a delete-d Jackknife sample will be $(n-d)$ and there will be $\binom{n}{d}$ of them.

8.3 Bootstrap Resampling

Suppose we have an i.i.d. sample from an unknown distribution, with cdf $F(\cdot)$. Denote the mean and variance of our unknown distribution μ and σ^2.

$$X_1, \ldots, X_n \overset{i.i.d.}{\sim} F(x).$$

We could estimate μ using the sample mean $\bar{X}$ as before; we may well be interested in the variance of this estimator. We know that

$$\text{Var}\left[\bar{X}\right] = \frac{\sigma^2}{n} = \frac{1}{n}\int (x-\mu)^2 F'(x)\,dx = \frac{1}{n}\sigma^2(F) = \beta(F)$$

and we also have

$$\mu = \int x\, F'(x)\,dx = \mu(F)$$

So we have that the variance of our estimator is a function of the unknown distribution function $F(x)$. To get around this inconvenience, we could consider simply replacing F by the empirical distribution function,

$$\widehat{F}(x) = \frac{i}{n} \qquad X_{(i)} \leq x < X_{(i+1)}$$

where $X_{(i)}$ is the i^{th} order statistic. Now, we can construct the following "bootstrap" estimators for μ and σ^2:

$$\widehat{\mu} = \mu(\widehat{F}) = \sum_{i=1}^{n} X_i \frac{1}{n} = \bar{X},$$

$$\widehat{\sigma^2} = \sigma^2(\widehat{F}) = \sum_{i=1}^{n} (X_i - \bar{X})^2 \frac{1}{n}. \quad \Rightarrow \widehat{\text{Var}}\left(\bar{X}\right) = \frac{1}{n^2}\sum_{i=1}^{n}(x_i-\bar{x})^2 = \beta(\widehat{F})$$

Note: this is the biased estimator for σ^2.

In general... suppose we have a fixed amount of data sampled from a population with cdf F, and we wish to estimate the parameter θ using the estimator

$$\widehat{\theta} = \widehat{\theta}(X_1, \ldots, X_n).$$

The variance for the estimator $\widehat{\theta}$ will in general be dependent on the data's unknown distribution function:

$$\text{Var}\left[\widehat{\theta}\right] = \beta(F),$$

for some function β.

Now, when assessing the accuracy of $\widehat{\theta}$ as an estimator for θ, we have two options:

- we could make assumptions on the shape of F, and deduce that the population distribution for θ will derive its shape from this...
- ...or we could use information contained in $\widehat{\theta}$ to approximate the population distribution; we use the sampling distribution $\widehat{F}$ as a proxy for the population distribution F; this is bootstrapping.

If the estimator $\widehat{\theta}$ is simple, as with the example on the previous page, then we may be able to obtain an explicit expression for $\beta(F)$ in terms of e.g. population moments. Then, we could plug in the estimate $\widehat{F}$ for F and obtain the **bootstrap estimator** for the variance of $\widehat{\theta}$:

$$\widehat{\text{Var}}_{boot}\left[\widehat{\theta}\right] = \beta(\widehat{F}),$$

using e.g. sample moments. In general, however, $\beta(\cdot)$ is not so simple...and we need to approximate $\beta(\widehat{F})$ through a simulation-based approach.

Ideally: If we were able to somehow obtain B i.i.d. n-length samples

$$X_1^{(b)}, \ldots, X_n^{(b)} \sim F, \qquad b = 1, \ldots, B,$$

then we could easily estimate the variance of any estimator $\hat{\theta}$ by creating

$$\hat{\theta}^{(b)} = \hat{\theta}\left(X_1^{(b)}, \ldots, X_n^{(b)}\right), \qquad b = 1, \ldots, B,$$

and calculating the sample variance of the $\hat{\theta}^{(b)}$. Indeed, we would be able to estimate the value of any integral with respect to $F'(x)$ ... this is Monte Carlo Integration/Estimation!

When bootstrapping: Given the data $X_1, \ldots, X_n \overset{i.i.d.}{\sim} F(x)$, construct an empirical estimate $\widehat{F}$ of the distribution function

$$\widehat{F}(x) = \frac{i}{n} \qquad X_{(i)} \leq x < X_{(i+1)}$$

where $X_{(i)}$ is the i^{th} order statistic. Then, given B i.i.d. samples from this empirical estimate,

$$X_1^{*(b)}, \ldots, X_n^{*(b)} \overset{i.i.d.}{\sim} \widehat{F}, \qquad b = 1, \ldots, B,$$

we define the **Monte Carlo approximation of the bootstrap estimator** for the variance of $\hat{\theta}$ to be

$$\widehat{\text{Var}}^*_{boot}\left[\widehat{\theta}\right] = \frac{1}{B-1}\sum_{b=1}^{B}\left(\widehat{\theta}^{*(b)} - \frac{1}{B}\sum_{k=1}^{B}\widehat{\theta}^{*(k)}\right)^2,$$

where

$$\widehat{\theta}^{*(b)} = \widehat{\theta}\left(X_1^{*(b)}, \ldots, X_n^{*(b)}\right), \qquad b = 1, \ldots, B.$$

- This methodology extends flexibly to estimating any distributional characteristics of $\widehat{\theta}$, not just the variance!

- The number of bootstrap samples, B, is not restricted by the length of the original dataset, unlike with the jackknife.
- Monte Carlo sampling from $\hat{F}$ ≡ resampling with replacement from the original data!
- Increasing B will reduce the MonteCarlo error, but not the error due to using $\hat{F}$ instead of F.