1. Basic Results — Inverse Transform Method

We want to use $\mathcal{U}(0,1)$ numbers to generate observations (variates) from other distributions.

Let X be a random variable with c.d.f. $F(\cdot)$. Then

$$U = F(X) \sim \mathcal{U}(0,1).$$

Proof: Let $Y = F(X)$ and suppose that Y has c.d.f. $G(y)$. Then (for the continuous case),

$$\begin{aligned} G(y) &= \mathsf{P}(Y \le y) = \mathsf{P}(F(X) \le y) \\ &= \mathsf{P}(X \le F^{-1}(y)) = F(F^{-1}(y)) \\ &= y. \quad \diamondsuit \end{aligned}$$

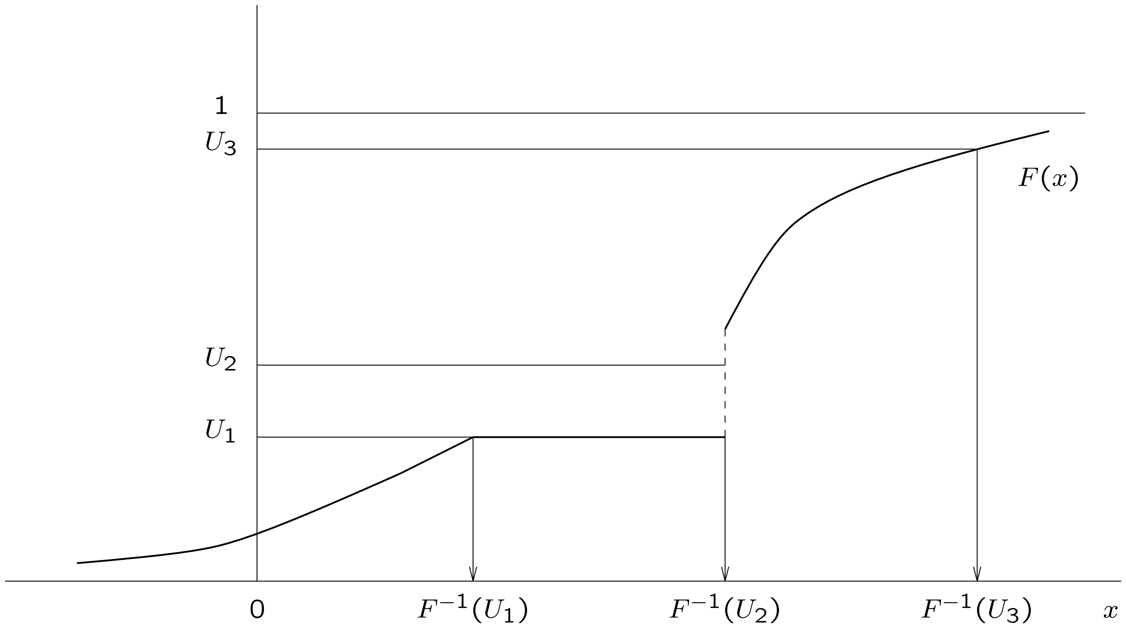
In the above, we defined the inverse c.d.f. by

$$F^{-1}(u) = \text{inf}[x : F(x) \geq u] \quad u \in [0, 1].$$

Let $U \sim \mathcal{U}(0, 1)$. Then the random variable $F^{-1}(U)$ has the same distribution as X.

1. Sample U from $\mathcal{U}(0, 1)$.

2. Return $X = F^{-1}(U)$.



1
U_3
$F(x)$
U_2
U_1
0
$F^{-1}(U_1)$
$F^{-1}(U_2)$
$F^{-1}(U_3)$
x

Example 1 Discrete Example. Suppose

$$X \equiv \begin{cases} -1 & \text{w.p. } 0.6 \\ 2.5 & \text{w.p. } 0.3 \\ 4 & \text{w.p. } 0.1 \end{cases}$$

x	Pr$(X = x)$	$F(x)$	$U(0,1)$'s
-1	0.6	0.6	[0.0,0.6]
2.5	0.3	0.9	(0.6,0.9]
4	0.1	1.0	(0.9,1.0]

Thus, if $U = 0.63$, we take $X = 2.5$.

Example 2 The $U(a,b)$ distribution.

$F(x) = (x-a)/(b-a)$, $a \leq x \leq b$.

Solving $(X-a)/(b-a) = U$ for X we get $X = a + (b-a)U$.

Example 3 The discrete uniform distribution on $\{a, a+1, \ldots, b\}$ with

$$\Pr(X = k) = \frac{1}{b-a+1}, \quad a, a+1, \ldots, b.$$

Clearly,

$$X = a + \lfloor (b-a+1)U \rfloor.$$

Example 4 The exponential distribution.

$F(x) = 1 - e^{-\lambda x}, x > 0.$

Solving $F(X) = U$ for X we get

$$X = -\frac{1}{\lambda}\ln(1-U) \quad \text{or} \quad X = -\frac{1}{\lambda}\ln U.$$

Example 5 The Weibull distribution

$F(x) = 1 - e^{-(\lambda x)^\beta}, x > 0.$

Solving $F(X) = U$ for X one has

$$X = \frac{1}{\lambda}\left[-\ln(1-U)\right]^{1/\beta} \quad \text{or} \quad X = \frac{1}{\lambda}\left[-\ln U\right]^{1/\beta}.$$

Example 6 The triangular distribution with density

$$f(x) = \begin{cases} x & \text{if } 0 \le x < 1 \\ 2 - x & \text{if } 1 \le x \le 2. \end{cases}$$

The c.d.f. is

$$F(x) = \begin{cases} x^2/2 & \text{if } 0 \le x < 1 \\ 1 - (2-x)^2/2 & \text{if } 1 \le x \le 2. \end{cases}$$

Notice that $F(1) = 1/2$. We have two cases:

(a) If $U < 1/2$, we solve $X^2/2 = U$ to get

$$X = \sqrt{2U}.$$

(b) If $U \ge 1/2$, the only root of $1 - (2 - X)^2/2 = U$ in $[1, 2]$ is

$$X = 2 - \sqrt{2(1-U)}.$$

Remark 7 Do not replace U by $1 - U$ here!

Example 8 The standard normal distribution. Unfortunately, the inverse c.d.f. $\Phi^{-1}(\cdot)$ does not have an analytical form. *This is often a problem with the inverse transform method.*

Easy solution: Do a table lookup. E.g., If $U = 0.975$, then $X = F^{-1}(U) = 1.96$.

More portable solution: The following approximation has absolute error $\leq 0.45 \times 10^{-3}$:

$$Z = \text{sign}(U - 1/2)\left(t - \frac{c_0 + c_1 t + c_2 t^2}{1 + d_1 t + d_2 t^2 + d_3 t^3}\right),$$

where $\text{sign}(x) = 1, 0, -1$ if x is positive, zero, or negative, respectively,

$$t = \{-\ln[\min(U, 1 - U)]^2\}^{1/2},$$

and

$$c_0 = 2.515517, \quad c_1 = 0.802853, \quad c_2 = 0.010328,$$
$$d_1 = 1.432788, \quad d_2 = 0.189269, \quad d_3 = 0.001308.$$

Example 9 The geometric distribution with probability function

$$\Pr(X = k) = (1-p)^{k-1}p, \quad k = 1, 2, \ldots$$

The c.d.f. is $F(k) = 1 - (1-p)^k, k = 1, 2, \ldots$

Hence,

$$\begin{aligned} X &= \min[k : 1 - (1-p)^k \geq U] \\ &= \left\lceil \frac{\ln(1-U)}{\ln(1-p)} \right\rceil. \end{aligned}$$

(Have to be a little careful about the "ceiling" function.)

3. Convolution Methods

Convolutions refer to adding things up.

Example 11 Binomial(n, p)

Suppose $X_1, \ldots, X_n$ are i.i.d. Bern(p). Then $Y = \sum_{i=1}^{n} X_i \sim$ Bin(n, p).

How do you get Bernoulli RV's?

Suppose $U_1, \ldots, U_n$ are i.i.d. U(0,1). If $U_i \leq p$, set $X_i = 1$; otherwise, set $X_i = 0$. Repeat for $i = 1, \ldots, n$.

Example 12 Erlang$_n(\lambda)$

Suppose $X_1, \ldots, X_n$ are i.i.d. Exp(λ). Then $Y = \sum_{i=1}^n X_i \sim$ Erlang$_n(\lambda)$.

Notice that by inverse transform,

$$\begin{aligned} Y &= \sum_{i=1}^n X_i \\ &= \sum_{i=1}^n \left[\frac{-1}{\lambda}\,\ell\mathrm{n}(U_i)\right] \\ &= \frac{-1}{\lambda}\,\ell\mathrm{n}\left(\prod_{i=1}^n U_i\right) \end{aligned}$$

(This only takes one natural log evaluation, so it's pretty efficient.)

Example 13 A simple Nor(0,1) generator.

Suppose that $U_1, \ldots, U_n$ are i.i.d. U(0,1), and let $Y = \sum_{i=1}^n U_i$.

Note that $\mathsf{E}[Y] = n/2$ and $\mathsf{Var}(Y) = n/12$.

By the CLT, for large n.

$$Y \;\approx\; \text{Nor}(n/2, n/12).$$

In particular, let's choose $n = 12$, and assume that it's "large". Then

$$Y - 6 \;=\; \sum_{i=1}^{12} U_i - 6 \;\approx\; \text{Nor}(0,1).$$

Crude, but effective!

By the way, if $Z \sim \text{Nor}(0,1)$ and you want $X \sim \text{Nor}(\mu, \sigma^2)$, just take $X \leftarrow \mu + \sigma Z$.

Other convolution-related tidbits:

Did you know?

$U_1 + U_2 \sim$ Triangular$(0, 1, 2)$.

If $X_1, \ldots, X_n$ are i.i.d. Geom(p), then $\sum_{i=1}^{n} X_i \sim$ NegBin(n, p).

If $Z_1, \ldots, Z_n$ are i.i.d. Nor(0,1), then $\sum_{i=1}^{n} Z_i^2 \sim \chi^2(n)$.

4. Acceptance-Rejection Method

Example 14 (Baby example, which you would usually do via inverse transform, but what the heck!)

Generate a U(2/3,1) RV. Here's the A-R algorithm:

1. Generate $U \sim \mathrm{U}(0,1)$.

2. If $U \geq 2/3$, ACCEPT $X \leftarrow U$. Otherwise, REJECT and go to 1.

Motivation: The majority of c.d.f.'s cannot be inverted efficiently.

Suppose we want to simulate a continuous RV with p.d.f. $f(x)$, but that it's difficult to generate directly. Also suppose that we can easily generate a RV having p.d.f. $h(x) \equiv t(x)/c$, where $t(x)$ *majorizes* $f(x)$, i.e.,

$$t(x) \geq f(x), \quad x \in \mathbb{R},$$

and

$$c \equiv \int_{-\infty}^{\infty} t(x)\,dx \geq \int_{-\infty}^{\infty} f(x)\,dx = 1,$$

where we assume that $c < \infty$.

Then f can be written as

$$f(x) = c \times \frac{f(x)}{t(x)} \times \frac{t(x)}{c} = cg(x)h(x),$$

where

$$\int_{-\infty}^{\infty} h(x)\, dx = 1 \quad (h \text{ is a density})$$

and

$$0 \leq g(x) \leq 1.$$

Theorem 15 (von Neumann, 1951) *Let $U \sim \mathcal{U}(0,1)$, and let Y a random variable with density h. If $U \leq g(Y)$, then Y has (conditional) density f.*

This suggests the following "acceptance-rejection" algorithm ...

Algorithm A-R

Repeat

Generate U from $\mathcal{U}(0,1)$

Generate Y from h

until $U \leq g(Y)$

Return $X \leftarrow Y$

There are two main issues:

- The ability to quickly sample from h.

- c must be small (t must be "close" to f) since

$$\Pr[U \leq g(Y)] = \frac{1}{c}$$

and the mean number of trials until "success" $[U \leq g(Y)]$ is equal to c.

Example 16 (Law & Kelton) Generate a RV with p.d.f. $f(x) = 60x^3(1-x)^2$, $0 \le x \le 1$. Can't invert this analytically.

Note that the maximum occurs at $x = 0.6$, and $f(0.6) = 2.0736$.

Using the majorizing function

$$t(x) = 2.0736, \quad 0 \le x \le 1$$

(which isn't actually too efficient), we get $c = \int_0^1 f(x)\,dx = 2.0736$, and therefore

$h(x) = 1, \quad 0 \le x \le 1 \quad$ (i.e., a U(0,1) p.d.f.)

and

$$g(x) = 60x^3(1-x)^2/2.0736.$$

E.g., if we generate $U = 0.13$ and $Y = 0.25$, then it turns out that $U \le g(Y) = \frac{60Y^3(1-Y)^2}{2.0736}$, so we take $X \leftarrow 0.25$.

Example 17 (Ross) The standard half-normal distribution with density

$$f(x) = \frac{2}{\sqrt{2\pi}} e^{-x^2/2}, \quad x \geq 0.$$

Using the majorizing function

$$t(x) = \sqrt{\frac{2e}{\pi}} e^{-x}$$

we get

$$c = \sqrt{\frac{2e}{\pi}} \int_0^\infty e^{-x}\, dx = \sqrt{\frac{2e}{\pi}} = 1.3155,$$

$$h(x) = e^{-x} \quad \text{[exponential}(\lambda = 1) \text{ density]},$$

and

$$g(x) = e^{-(x-1)^2/2}.$$

How can we generate from $N(0,1)$?

Generate U from $\mathcal{U}(0,1)$

Generate X from the half-normal distribution

Return

$$Z = \begin{cases} -X & \text{if } U \le 1/2 \\ X & \text{if } U > 1/2. \end{cases}$$

How can we generate from $N(\mu, \sigma^2)$?

Use the transformation $\mu + \sigma Z$.

Example 18 The gamma distribution with density

$$f(x) = \frac{(x/\alpha)^{\beta-1}}{\alpha\Gamma(\beta)} e^{-(x/\alpha)^{\beta}}, \quad x > 0.$$

If the shape parameter $\beta < 1$, we use the following A-R algorithm with $c \leq 1.39$:

Algorithm GAM1

$b \leftarrow (e+\beta)/e$ (e is the base of the natural logarithm)

While (True)

Generate U from $\mathcal{U}(0,1)$; $W \leftarrow bU$

If $W < 1$

$Y \leftarrow W^{1/\beta}$; Generate V from $\mathcal{U}(0,1)$

If $V \le e^{-Y}$: Return $X = \alpha Y$

Else

$Y \leftarrow -\ln[(b-W)/\beta]$

Generate V from $\mathcal{U}(0,1)$

If $V \le Y^{\beta-1}$: Return $X = \alpha Y$

If $\beta \geq 1$, the value of c for the following A-R algorithm decreases from $4/e = 1.47$ to $\sqrt{4/\pi} = 1.13$ as β increases from 1 to ∞.

Algorithm GAM2

$a \leftarrow (2\beta - 1)^{-1/2}$; $b \leftarrow \beta - \ln 4$; $c \leftarrow \beta + a^{-1}$; $d \leftarrow 1 + \ln 4.5$

While (True)

Generate U_1, U_2 from $\mathcal{U}(0, 1)$

$V \leftarrow a \ln[U_1/(1 - U_1)]$

$Y \leftarrow \beta e^V$; $Z \leftarrow U_1^2 U_2$

$W \leftarrow b + cV - Y$

If $W + d - 4.5Z \geq 0$: Return $X = \alpha Y$

Else

If $W \geq \ln Z$: Return $X = \alpha Y$

Example 19 The Poisson distribution with probability function

$$\Pr(X = n) = e^{-\lambda}\,\frac{\lambda^n}{n!}, \quad n = 0, 1, \ldots$$

Define A_i as the ith interarrival time from a Pois(λ) process. Then

$$\begin{aligned}
&X = n \\
&\quad\Leftrightarrow \quad \text{See exactly } n \text{ PP}(\lambda) \text{ arrivals by } t = 1 \\
&\quad\Leftrightarrow \quad \sum_{i=1}^{n} A_i \le 1 < \sum_{i=1}^{n+1} A_i \\
&\quad\Leftrightarrow \quad \sum_{i=1}^{n} \left[\frac{-1}{\lambda}\ell\mathsf{n}(U_i)\right] \le 1 < \sum_{i=1}^{n+1} \left[\frac{-1}{\lambda}\ell\mathsf{n}(U_i)\right] \\
&\quad\Leftrightarrow \quad \frac{-1}{\lambda}\,\ell\mathsf{n}\left(\prod_{i=1}^{n} U_i\right) \le 1 < \frac{-1}{\lambda}\,\ell\mathsf{n}\left(\prod_{i=1}^{n+1} U_i\right) \\
&\quad\Leftrightarrow \quad \prod_{i=1}^{n} U_i \ge e^{-\lambda} > \prod_{i=1}^{n+1} U_i. \qquad (5)
\end{aligned}$$

The following A-R algorithm samples U(0,1)'s until (5) becomes true.

Algorithm POIS1

$a \leftarrow e^{-\lambda}$; $p \leftarrow 1$; $X \leftarrow -1$

Until $p \leq a$

Generate U from $\mathcal{U}(0,1)$

$p \leftarrow pU$; $X \leftarrow X + 1$

Return X

Example 20 Apply Algorithm POIS1 to obtain a Pois($\lambda = 2$) variate.

Sample until $e^{-\lambda} = 0.1353 > \prod_{i=1}^{n+1} U_i$.

n	U_{n+1}	$\prod_{i=1}^{n+1} U_i$	Stop?
0	0.3911	0.3911	No
1	0.9451	0.3696	No
2	0.5033	0.1860	No
3	0.7003	0.1303	Yes

Thus, we take $X = 3$.

Remark 21 How many U's are required to generate one realization of X? Easy argument says that the expected number you'll need is E$[X+1] = \lambda + 1$.

Algorithm POIS2 (For $\lambda \geq 20$)

$a \leftarrow \pi\sqrt{\lambda/3}$; $b \leftarrow a/\lambda$; $c \leftarrow 0.767 - 3.36/\lambda$; $d \leftarrow \ln c - \ln b - \lambda$

Repeat

Repeat

Generate U from $\mathcal{U}(0,1)$

$Y \leftarrow [a - \ln((1-U)/U]/b$

until $Y > -1/2$

$X \leftarrow \lfloor Y + 1/2 \rfloor$

Generate V from $\mathcal{U}(0,1)$

until $a - bY + \ln[V/(1+e^{a-bY})^2] \leq d + X \ln \lambda - \ln(X!)$

Return X

Alternatively, we can use the normal approximation

$$\frac{X-\lambda}{\sqrt{\lambda}} \approx N(0,1).$$

Algorithm POIS3 (For $\lambda \geq 20$)

$\alpha \leftarrow \sqrt{\lambda}$

Generate Z from $N(0,1)$

Return $X = \max(0, \lfloor \lambda + aZ + 0.5 \rfloor)$ (Note that this employs a "continuity correction."

6. Generating Poisson Arrivals

When the arrival rate is constant, say λ, the interarrival times are i.i.d. exponential(λ) and the arrival times are generated recursively:

$$\begin{aligned} T_0 &= 0 \\ T_i &= T_{i-1} - \frac{1}{\lambda} \ln U_i, \quad i \geq 1 \end{aligned}$$

How can we generate a fixed number n of arrivals in a time interval $[a,b]$?

Generate $U_1, \ldots, U_n$ from $\mathcal{U}(0,1)$

Sort the U_i's: $U_{(1)} < U_{(2)} < \cdots < U_{(n)}$

Set the arrival times to $T_i = a + (b-a)U_{(i)}$

7. Special-Case Techniques

7.1 Box-Müller Method

Nice way to generate standard normals.

Theorem 23 *If* U_1, U_2 *are i.i.d. U(0,1), then*

$$\begin{aligned} Z_1 &= \sqrt{-2\ell\text{n}(U_1)}\cos(2\pi U_2) \\ Z_2 &= \sqrt{-2\ell\text{n}(U_1)}\sin(2\pi U_2) \end{aligned}$$

are i.i.d. Nor(0,1).

Note that the trig calculations must be done in radians.

Proof Someday soon. $\diamondsuit$

Some cool corollaries from Box-Müller.

Example 24 Note that

$$Z_1^2 + Z_2^2 \sim \chi^2(1) + \chi^2(1) \sim \chi^2(2).$$

But

$$\begin{aligned}
& Z_1^2 + Z_2^2 \\
& \quad = -2\ell\mathrm{n}(U_1)(\cos^2(2\pi U_2) + \sin^2(2\pi U_2)) \\
& \quad = -2\ell\mathrm{n}(U_1) \\
& \quad \sim \mathrm{Exp}(1/2).
\end{aligned}$$

Thus, we've just proven that

$$\chi^2(1) + \chi^2(1) \sim \mathrm{Exp}(1/2).$$

Example 25 Note that

$$Z_1/Z_2 \sim \text{Nor}(0,1)/\text{Nor}(0,1) \sim \text{Cauchy}.$$

But

$$\begin{aligned} Z_1/Z_2 &= \frac{\sqrt{-2\ell\text{n}(U_1)}\,\sin(2\pi U_2)}{\sqrt{-2\ell\text{n}(U_1)}\,\cos(2\pi U_2)} \\ &= \tan(2\pi U_2). \end{aligned}$$

Thus, we've just proven that

$$\tan(2\pi U) \sim \text{Cauchy}.$$

Similarly,

$$\cot(2\pi U) \sim \text{Cauchy}.$$

Similarly,

$$Z_1^2/Z_2^2 = \tan^2(2\pi U) \sim F(1,1).$$

(Did you know that?)

Polar Method — a little faster than Box-Müller:

1. Generate U_1, U_2 i.i.d. U(0,1).

Let $V_i = 2U_i - 1$, $i = 1, 2$, and $W = V_1^2 + V_2^2$.

2. If $W > 1$, reject and go back to step 1.

Otherwise, let $Y = \sqrt{-2(\ell\mathrm{n} W)/W}$, and accept $Z_i \leftarrow V_i Y$, $i = 1, 2$.

7.2 Order Statistics

Suppose that $X_1, X_2, \ldots, X_n$ are i.i.d. from some distribution, and let $Y = \min\{X_1, \ldots, X_n\}$. (Y is called the first order stat.) Can you generate Y using just *one* U(0,1)?

Example 26 Suppose $X_1, \ldots, X_n \sim \text{Exp}(\lambda)$. Then

$$\begin{aligned}
\Pr(Y \leq y) &= 1 - \Pr(Y > y) \\
&= 1 - \Pr(\min_i X_i > y) \\
&= 1 - \Pr(\text{all } X_i\text{'s} > y) \\
&= 1 - (e^{y})^n.
\end{aligned}$$

This implies that $Y = \min_i\{X_i\} \sim \text{Exp}(n\lambda)$. So you can generate

$$Y = -\frac{1}{n\lambda}\ell\text{n}(U).$$

Can you do the same kind of thing for $Z = \max_i X_i$?

7.3 Other Quickies

$\chi^2(n)$ distribution: If $Z_1, Z_2, \ldots, Z_n$ are i.i.d. N(0,1), then $\sum_{i=1}^n Z_i^2 \sim \chi^2(n)$.

$t(n)$ distribution: If $Z \sim$ N$(0,1)$ and $Y \sim \chi^2(n)$, and X nd Y are independent, then

$$\frac{Z}{\sqrt{Y/n}} \sim t(n).$$

Note that $t(1)$ is the Cauchy distribution.

$F(n,m)$ distribution: If $X \sim \chi^2(n)$ and $Y \sim \chi^2(m)$ and X and Y are independent, then $(X/n)/(Y/m) \sim F(n,m)$.

Generating RV's from continuous empirical distributions — no time here. Can use the CONT function in Arena.

8. The Multivariate Normal Distribution

The random vector $\mathbf{X} = (X_1, \ldots, X_k)^\top$ has the multivariate normal distribution with mean vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_k)^\top$ and $k \times k$ covariance matrix Σ if, $\forall \mathbf{x} \in \mathbb{R}^k$, it has p.d.f.

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{k/2} |\Sigma|^{1/2}} \exp\left\{ -\frac{(\mathbf{x} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})}{2} \right\}.$$

It turns out that

$$E(X_i) = \mu_i, \quad \mathrm{Var}(X_i) = \sigma_{ii}, \quad \mathrm{Cov}(X_i, X_j) = \sigma_{ij}.$$

It can be shown that $\Sigma = CC^\top$, where C is lower triangular, and

$$\mathbf{Z} = C^{-1}(\mathbf{X} - \boldsymbol{\mu}) \sim N(\mathbf{0}, I).$$

The following algorithm computes C (Cholesky):

Algorithm LTM (amended by DG, 4/06)

For $i = 1, \ldots, k$,

For $j = 1, \ldots, i-1$,

$$c_{ij} \leftarrow \left(\sigma_{ij} - \sum_{\ell=1}^{j-1} c_{i\ell} c_{j\ell}\right) / c_{jj}$$

$$c_{ji} = 0$$

$$c_{ii} = \left(\sigma_{ii} - \sum_{\ell=1}^{i-1} c_{i\ell}^2\right)^{1/2}$$

Once C has been computed, $\mathbf{X}$ is generated as follows:

Algorithm MN

$i \leftarrow 1$

Until $i > k$:

  Generate X_i from $N(0,1)$

  $X_i \leftarrow \mu_i$

  $j \leftarrow 1$

  Until $j > i$: $X_i \leftarrow X_i + c_{ij}X_j$; $j \leftarrow j + 1$

  $i \leftarrow i + 1$

9. Some Stochastic Processes

9.1 First-Order Moving Average Process

An MA(1) process is defined by

$$Y_i = \varepsilon_i + \theta\varepsilon_{i-1}, \quad \text{for } i = 1, 2, \ldots,$$

where the ε_i's are i.i.d. Nor$(0, 1)$ random variables that are independent of Y_0.

The MA(1) has covariance function Var$(Y_i) = 1 + \theta^2$, Cov$(Y_i, Y_{i+1}) = \theta$, and Cov$(Y_i, Y_{i+k}) = 0$ for $k \geq 2$.

So the covariances die off pretty quickly.

How to generate? Start with $\varepsilon_0 \sim$ Nor$(0, 1)$. Then generate $\varepsilon_1 \sim$ Nor$(0, 1)$ to get Y_1, $\varepsilon_2 \sim$ Nor$(0, 1)$ to get Y_2, etc.

The MA(1) is a popular tool for modeling detecting trends.

9.2 First-Order Autoregressive Process

An AR(1) process is defined by

$$Y_i = \phi Y_{i-1} + \varepsilon_i, \quad \text{for } i = 1, 2, \ldots,$$

where $-1 < \phi < 1$; Y_0 is a Nor$(0,1)$ random variable; and the ε_i's are i.i.d. Nor$(0, 1-\phi^2)$ random variables that are independent of Y_0.

The AR(1) has covariance function $\text{Cov}(Y_i, Y_{i+k}) = \phi^{|k|}$ for all $k = 0, \pm 1, \pm 2, \ldots$.

If ϕ is close to one, you get highly positively correlated Y_i's. If ϕ is close to zero, the Y_i's are nearly independent.

How to generate? Start with $Y_0 \sim$ Nor$(0,1)$ and $\varepsilon_1 \sim \sqrt{1-\phi^2}$ Nor$(0,1)$. This gives Y_1. Then generate ε_2 to get Y_2, etc.

This is used to model lots of real-world stuff.

9.3 M/M/1 Queue.

Consider a single-server queue with customers arriving according to a Poisson(λ) process, standing in line with a FIFO discipline, and then getting served in an Exp(μ) amount of time. Let I_{i+1} denote the interarrival time between the ith and $(i+1)$st customers; let S_i be the ith customer's service time; and let W_i denote the ith customer's waiting time before service. Lindley gives a very nice way to generate a series of waiting times for this simple example:

$$W_{i+1} \; = \; \max\{W_i - S_i + I_{i+1},\, 0).$$

(Can you model time in system with a similar equation?)

9.4 Brownian Motion.

The stochastic process $\{\mathcal{W}(t), t \geq 0\}$ is a *standard Brownian motion* process if:

(a) $\mathcal{W}(0) = 0$.

(b) $\mathcal{W}(t) \sim \text{Nor}(0, \sigma^2 t)$.

(c) $\{\mathcal{W}(t), t \geq 0\}$ has stationary and indep increments.

Increments: Anything like $\mathcal{W}(b) - \mathcal{W}(a)$.

Stationary increments: The distribution of $\mathcal{W}(t+h) - \mathcal{W}(t)$ only depends on h.

Independent increments: If $a < b < c < d$, then $\mathcal{W}(d) - \mathcal{W}(c)$ is indep of $\mathcal{W}(b) - \mathcal{W}(a)$.

Discovered by Brown; first analyzed rigorously by Einstein; mathematical rigor established by Wiener (also called *Wiener* process).

Here's a way to construct BM:

Suppose $Y_1, Y_2, \ldots$ is any sequence of identically distributed RV's with mean zero and variance 1. (To some extent, the Y_i's don't even have to be indep!) *Donsker's* Central Limit Theorem says that

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} Y_i \xrightarrow{d} \mathcal{W}(t) \quad \text{as } n \to \infty,$$

where $\xrightarrow{d}$ denotes convergence in distribution as n gets big, and $\lfloor \cdot \rfloor$ means to round down to the next integer, e.g., $\lfloor 3.7 \rfloor = 3$.

One choice that works well is to take $Y_i = \pm 1$, each with probability 1/2. Take n at least 100, $t = 1/n, 2/n, \ldots, n/n$, and calculate $\mathcal{W}(1/n), \mathcal{W}(2/n), \ldots, \mathcal{W}(n/n)$.

It really works!