Chapter 3

General Sampling Methods
--- Generation of Nonuniform Random Variables

Generation of Nonuniform Random Variables

- How to generate samples from an arbitrary statistical distribution?
- We are given the CDF $F(x)$, and we want to generate random variates distributed according to $F(x)$.
- For a continuous RV with a density $f(x)$, there are two general methods:
- **Inversion and**
- **Acceptance–rejection (A–R).**
- There are a variety of more ad hoc methods that use special properties of the target distribution.

Generation of Nonuniform Random Variables

If your distribution has a name

Normal, exponential, binomial, Poisson, etc.

then it is probably already in

R, Python, Matlab, Julia, Mathematica, etc.

We will look briefly because

- Sometimes a new distribution comes up
- The same ideas get used later

1. Inverse Transform Method

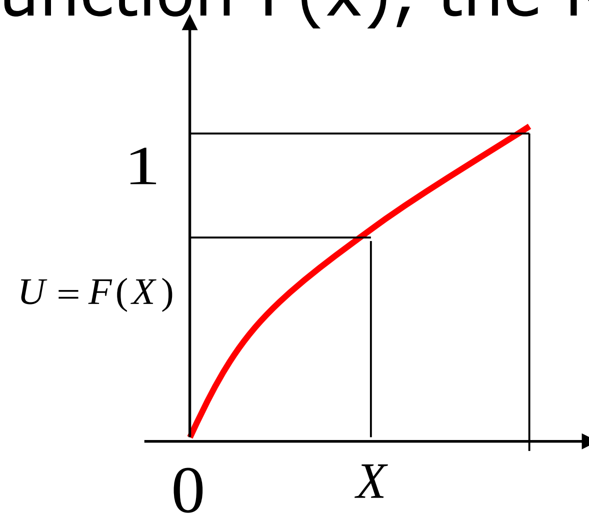
Goal: Generate samples from a distribution F(x).

Theorem:

Let U be a uniform (0,1) R.V. For any continuous and strictly increasing distribution function F(x), the R.V. defined by

$$X := F^{-1}(U)$$

has distribution function F(x).



Proof.

$$F_X(x) = P(X \le x) = P(F^{-1}(U) \le x) = P(U \le F(x)) = F(x).$$

1. Inverse Transform Method

- **Generate samples from a distribution F(x).**

- **Algorithm:**

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

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

  (or set F(X) = U, and solve for X)

Ex: (exponential distribution)

Let X has density

$$f(x) = \lambda e^{-\lambda x}, \; x \geq 0. \; E(X) = \frac{1}{\lambda}.$$

Its distribution function is

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

Then
$$X = F^{-1}(U) = -\frac{1}{\lambda}\log(1-U).$$

Note U and 1-U has the same distribution.
So we may let

$$X = F^{-1}(U) = -\frac{1}{\lambda}\log(U).$$

Ex: Chi-square distribution:

- Note: Relations among different distributions can be used to generate random numbers.
- The exponential distribution with parameter $\lambda = 1/2$ is a Gamma distribution Gamma(1, 2)
- Chi-square distribution $\chi^2(2n)$ with degree of freedom 2n is a Gamma distribution Gamma (n, 2).
- Using this relation one has a generation method for Chi-square distribution

$$X = -2\sum_{i=1}^{n} \log(U_i) \sim \chi^2(2n).$$

(Note: Gamma distribution has **additive property**)

Ex: Uniform distribution on [a, b]

- The distribution function is

$$F(x)=\begin{cases}0, & x<a,\\ \dfrac{x-a}{b-a}, & a\le x<b,\\ 1, & x\ge b.\end{cases}$$

We set

$$\frac{X-a}{b-a}=U$$

and solve for X: X = a + (b-a) U.

Ex (Rayleigh distribution)

- Let

$$X \sim F(x) = 1 - e^{-2x(x-b)}, \; x \geq b.$$

Solving the equation F(x) = u, 0<u<1, results in

$$x = \frac{b}{2} + \frac{\sqrt{b^2 - 2\log(1-u)}}{2}.$$

Then Rayleigh distribution can be generated as

$$X = \frac{b}{2} + \frac{\sqrt{b^2 - 2\log(U)}}{2}, \quad U \sim U(0,1).$$

1. Inverse Transform Method

Name	Density	Distribution function	Random variate
Exponential	$e^{-x}, x > 0$	$1 - e^{-x}$	$\log(1/U)$
Weibull (a), $a > 0$	$ax^{a-1}e^{-x^a}, x > 0$	$1 - e^{-x^a}$	$(\log(1/U))^{1/a}$
Gumbel	$e^{-x}e^{-e^{-x}}$	$e^{-e^{-x}}$	$-\log\log(1/U)$
Logistic	$\frac{1}{2+e^x+e^{-x}}$	$\frac{1}{1+e^{-x}}$	$-\log((1-U)/U)$
Cauchy	$\frac{1}{\pi(1+x^2)}$	$1/2 + (1/\pi)\arctan x$	$\tan(\pi U)$
Pareto (a), $a > 0$	$\frac{a}{x^{a+1}}, x > 1$	$1 - 1/x^a$	$1/U^{1/a}$

Table 1: Some densities with distribution functions that are explicitly invertible.

Remark

Sometime, the inverse of F is not available in explicit form, the inverse transform method is still applicable through **numerical evaluation** of the inverse of F.

For example, Newtow's method can be used to find the root of the equation F(X) = U:

$$x_{n+1} = x_n - \frac{F(x_n) - U}{f(x_n)}.$$

Given a starting point x_0, compute x_n up to the step at which $|x_{(n+1)} - x_n|$ is less than a prescribed threshold, and set $X = x_{(n+1)}$.

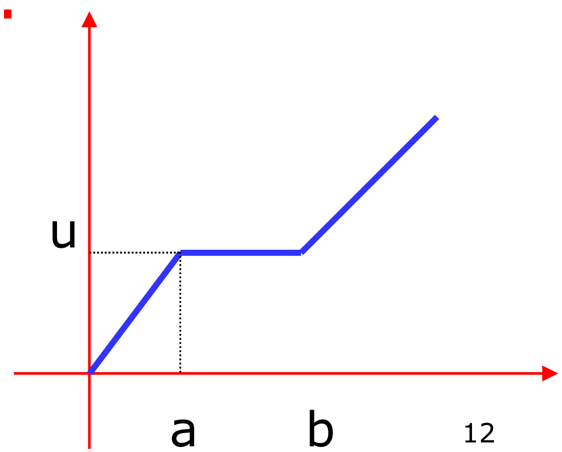
Remark

The inverse of F is well defined when F is strictly increasing. Otherwise, we may need a rule to break the ties. For example, set

$$X = F^{-1}(U) := \inf \{X : F(X) \geq U\}.$$

If there are many values of x for which F(x)=u, the definition chooses the **smallest**.

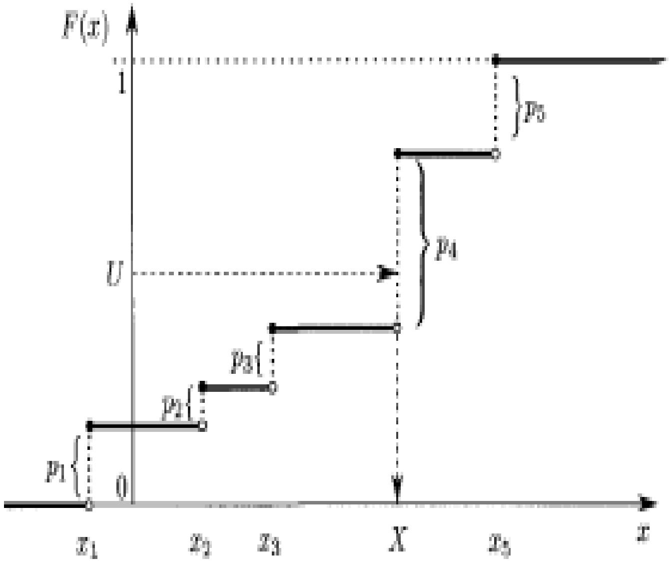
The Theorem above is still true.



Discrete Inverse-Transform Method

For a discrete random variable

$$X \sim \begin{pmatrix} x_1 \cdots\cdots x_n \\ p_1 \cdots\cdots p_n \end{pmatrix}, \quad x_1 < \cdots < x_n, \sum_{i=1}^{n} p_i = 1.$$



(1) Generate U~ U(0,1) ;

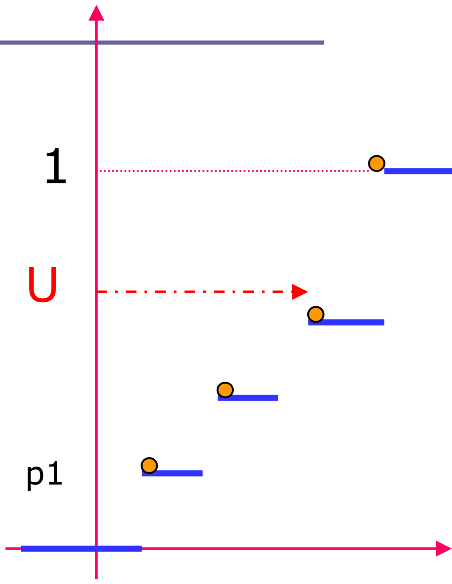
(2) Find the smallest positive integer k such that

$$\boldsymbol{F(x_k) \geq U} \textbf{ and return } \mathbf{X} = \boldsymbol{x_k}.$$

Discrete Inverse-Transform Method

- Discrete distribution:

$$X \sim \begin{pmatrix} x_1 \cdots\cdots x_n \\ p_1 \cdots\cdots p_n \end{pmatrix}, \quad x_1 < \cdots < x_n, \sum_{i=1}^{n} p_i = 1.$$



To sample from this discrete distribution:

(1) Generate a uniform U;

(2) Find k in {1, …, n}, such that

$$F(x_{k-1}) = p_1 + \cdots + p_{k-1} < U \le p_1 + \cdots + p_k = F(x_k),$$

(3) Set X = x_k.

This x_k is the smallest one such that $F(x_k) \ge U.$

Ex: Bernoulli distribution X~B(1,p), i.e., $X \sim \begin{pmatrix} 0 & 1 \\ p & 1-p \end{pmatrix}$.

To sample from Bernoulli distribution:

(1) Generate a uniform U;

(2) If U <= p, set X=0;

otherwise, set X=1.

Ex: 二项分布 **X ~ B(n, p),**

$$P(X=k)=\begin{pmatrix} n \\ k \end{pmatrix} p^k (1-p)^{n-k}, k=0,1,\cdots,n.$$

$$\Rightarrow X = B_1 + B_2 + \cdots + B_n, B_i \sim B(1,p), \textbf{i.i.d.}$$

Ex

- Suppose we wish to draw N = 10^5 independent copies of a discrete RV taking values 1, … , 5 with probabilities 0.2, 0.3, 0.1, 0.05, 0.35, respectively.
- The following MATLAB program implements the inverse transform method to achieve this, and records the frequencies of occurrences of 1,…,5.

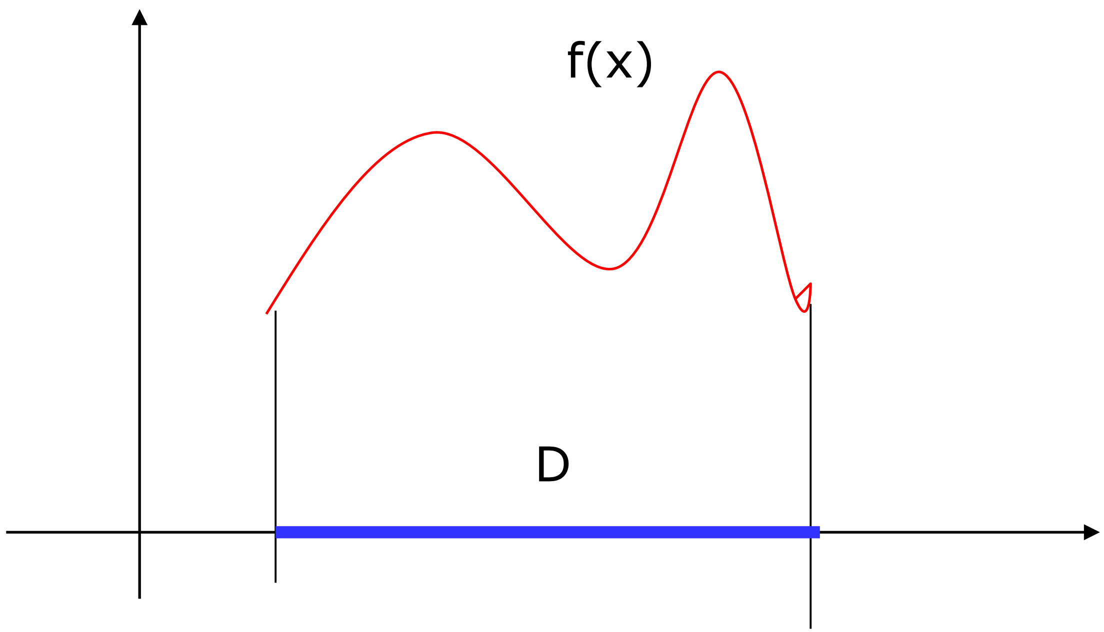
```
p = [0.2,0.3,0.1,0.05,0.35];
N = 10^5;
x = zeros(N,1);
for i=1:N
   x(i) = min(find(rand<cumsum(p))); %draws from p
end
freq = hist(x,1:5)/N
```

2. Acceptance-Rejection Method

- **The simplest version:**

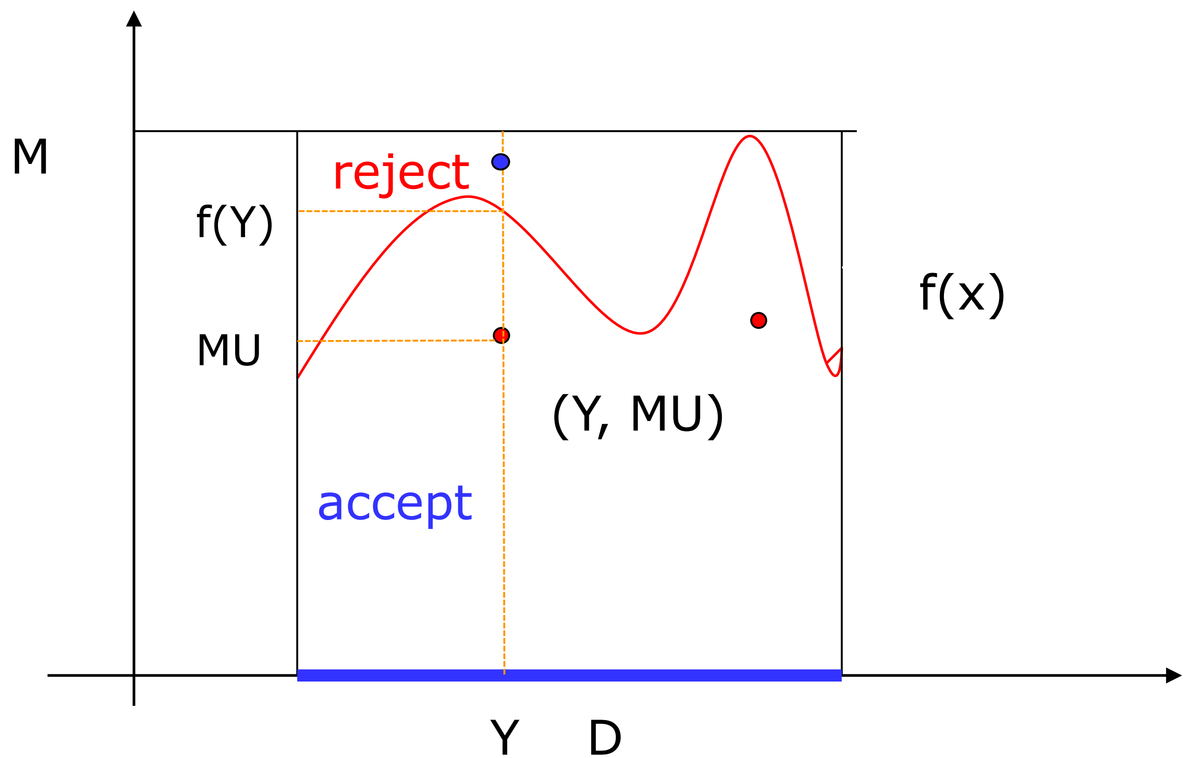
  The goal is to sample from a density $f(x),\ x \in D.$

  The support D of f is bounded, let g(·) be the uniform distribution on D.



2. Acceptance-Rejection Method

Generate points (Y, MU) uniformly on the square



If the point falls below the curve of f(x), then accept Y.

2. Acceptance-Rejection Method

The goal is to sample from a density $f(x),\ x \in D.$

Let g(·) be the uniform distribution on D.

Choose c, such that c g(·) = max f(x)=:M. Then

$$f(x) \le cg(x) =: M.$$

1. Generate Y from uniform dis. on D, i.e., from g(·);
2. Generate U from U(0,1), independent from Y.
3. If $M\,U \le f(Y)$, set X = Y;
   otherwise, go to Step 1.

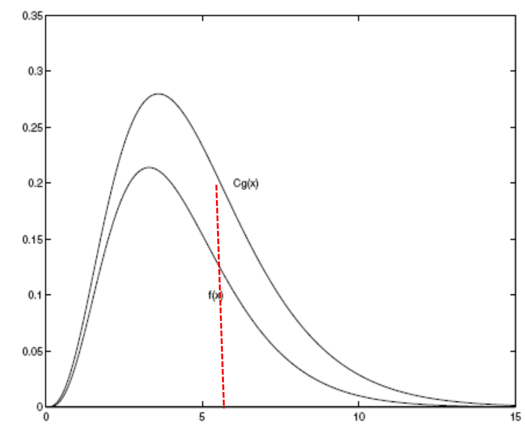
Efficiency = 1/c.

If M if large, then c is large, the efficiency is low.

General Case

Let $X \sim f(x),\ x \in D$. Let g(·) be a density (with the same support D) from which we know how to generate samples and (for some constant c)

$$f(x) \le c\, g(x),\ x \in D.$$



The goal is to sample from f(x):

1. Generate Y from g(·);
2. Generate U from U(0,1), independent from Y.
3. If $U \le \dfrac{f(Y)}{cg(Y)}$, set X=Y;

   otherwise, go to Step 1.

Given Y, the probability of accept this point is
f(Y)/(cg(Y))

Theorem: **The random variable generated by the acceptance-rejection method has density f(x).**

Proof: $$P(Y \le x \text{ and } Y \text{ is accepted}) = P\left(Y \le x, U \le \frac{f(Y)}{cg(Y)}\right)$$

$$= \int_{-\infty}^{x} \int_{0}^{\frac{f(y)}{cg(y)}} g(y)\,du\,dy = \int_{-\infty}^{x} \frac{f(y)}{cg(y)} g(y)\,dy$$

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

F(x) is the CDF

$$P(Y \text{ is accepted}) = \frac{1}{c} \quad (\text{set } x = +\infty \text{ above})$$

$$\Rightarrow P(Y \le x \mid Y \text{ is accepted}) = \frac{c^{-1}F(x)}{c^{-1}} = F(x).$$

2. Acceptance-Rejection Method

- The function g(·) is called the **proposal** pdf. Assume it is easy to generate samples from it.
- **The efficiency is defined as the probability of acceptance:**

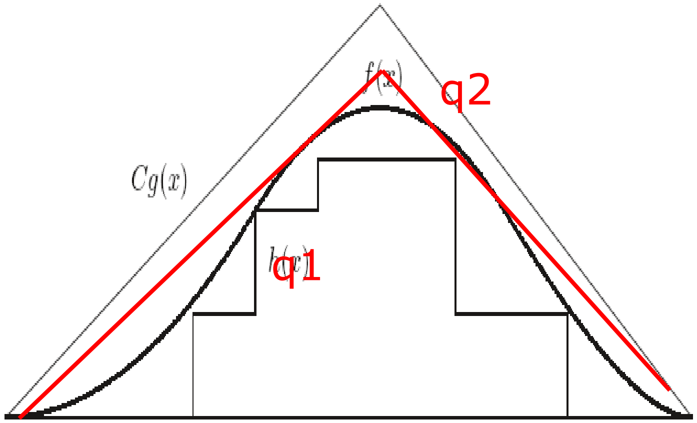
$$P(Y \textbf{ is accepted}) = 1/c.$$

- Since the trials are independent, the number of trials required to obtain a successful pair has geometric distribution: Ge (1/c). Thus the expected number of trials is equal to C.
- The AR method is one of the most useful general methods for sampling from general distributions.

Remark:

If the function f is too time-consuming to evaluate, we can use squeeze functions q1 and q2 with

$$q_1(x) \le f(x) \le q_2(x) \le c\, g(x),\ x \in D$$



$If\ \ U \le q_1(Y)/(cg(Y))$, **then y can immediately be accepted;**

$U > q_2(Y)/(cg(Y))$, **then y can immediately be rejected;**

Only if both cases do not apply, should function f be evaluated.

Example

$$f(x) = 30(x^2 - 2x^3 + x^4),\ x \in (0,1)$$

In this example, g(x) can be chosen as g(x)=1 for x in (0,1), and c can be chosen as

$$c = \max_{x \in (0,1)} f(x) = 30/16.$$

Efficiency: 1/c = 16/30.

The algorithm is as follows

(1) Draw U_1 and U_2 form U(0,1), independent;

(2) If

$$cU_2 \leq f(U_1), i.e.,$$

$$U_2 \leq 16(U_1^2 - 2U_1^3 + U_1^4)$$

accept U_1,

otherwise, reject and go back to step 1.

Efficiency:

The average number of iteration to generate one random variable is 30/16.

Remarks:

- There are many variations of the acceptance rejection methods. The methods described above uses a sequences of **i.i.d.** variates from $g(\cdot)$.
- A method using a **non-independent** sequence is called **Metropolis method (MCMC)**.
- The AR method can also be used for **multivariate** distribution (but the efficiency can be low).

Example

Generate from the **Positive normal**

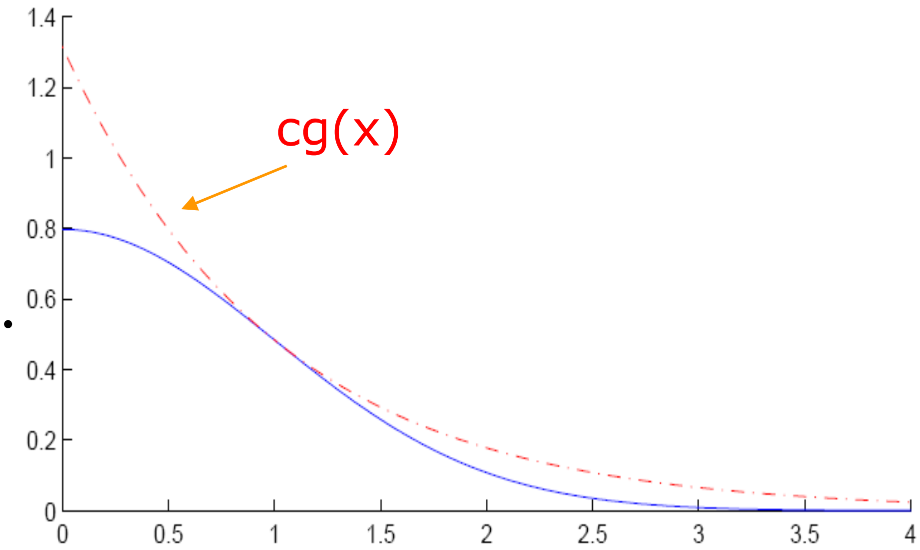
$$f(x) = \sqrt{\frac{2}{\pi}} e^{-x^2/2}, x \geq 0.$$

Choose g(x) $= e^{-x}$ **– pdf of Exp (1).**

The smallest constant C such that

$f(x) \leq Cg(x)$ **is** $\sqrt{2e/\pi}$.

The efficiency is $\sqrt{\pi / 2e} \approx 0.76.$



3. The Composition Method

The probability function F (x) is a **mixture**:

$$F(x) = \sum_{j=1}^{r} p_j F_j,\ p_j > 0, \sum_{j=1}^{r} p_j = 1$$

$$\textbf{or}\ \ f(x) = \sum_{j=1}^{r} p_j f_j,\ p_j > 0, \sum_{j=1}^{r} p_j = 1,$$

where F_j or f_j are distribution functions or density functions. In this case, we may:

(1) Sample j from the discrete distribution

$$\eta \sim \begin{pmatrix} 1 & \cdots\cdots r \\ p_1 & \cdots\cdots p_r \end{pmatrix},$$

(2) Sample X from the distribution F_j or f_j.

Theorem:

The random variable generated by the composition method has the required distribution function F(x).

Proof.

$$P(X \le x) = \sum_{j=1}^{r} P(X \le x \mid \eta = j)\ P(\eta = j)$$

$$= \sum_{j=1}^{r} F_j(x) p_j = F(x).$$

Ex

Let X ~ F(x), with

$$F(x) = \sum_{j=1}^{\infty} c_j x^j, \ 0 < x < 1, c_j > 0, \sum_{j=1}^{\infty} c_j = 1.$$

Then the samples of X can be generated as:

(1) Generate two independent uniform U_1 and $U_2 \sim U(0,1)$;

(2) If $\sum_{j=1}^{k-1} c_j < U_1 \leq \sum_{j=1}^{k} c_j$ (sampling from discrete dis.)

set X = $(U_2)^{1/k}$ (inverse transform method for F_k)

Remark

- Each of the two steps themselves require some generating method to be used, for instance inversion based on two independent uniform numbers U_1 and U_2 (one for generating j, the other one for X).
- Note also that unlike inversion, we need at least two uniform numbers to generate one variate.
- The composition method arises naturally for **mixture distributions**, but it can also be useful to tackle complicated density functions by breaking them down into different components.

Remark

- In the case where it is not easy to generate random draws from *f(x),* we may approximate f(x) as a weighted sum of $f_1(x)$, f2(x) , …, $f_r(x)$, where $f_i(x)$ can take any distribution function.

4. Normal Variates

- **Box-Muller method:**

$$X = \sqrt{-2\ln U}\,\sin(2\pi V),$$

$$Y = \sqrt{-2\ln U}\,\cos(2\pi V),$$

where U, V are independent uniform in (0,1)x(0,1).

We may proof that the joint distribution for X, Y is $N(0, I_2)$.

Theorem:

Let (X, Y) be independent standard normal variates, and let

$$\begin{cases} X = R\cos\Theta, \\ Y = R\sin\Theta. \end{cases}$$

Then R, Θ are independent random variables, R^2 is the exponential variable with mean 2, and Θ has the uniform distribution on (0, 2π) . They can be generated as

$$R = \sqrt{-2\ln U},$$

$$\Theta = 2\pi V.$$

Proof (outline)

The joint distribution of R, Θ is

$$q(r,\theta) = p(x,y)\,|J| = \frac{1}{2\pi} e^{-r^2/2}\, r,\ 0 \le \theta \le 2\pi, 0 < r < \infty,$$

where $J = \begin{vmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{vmatrix} = r\cos^2\theta + r\sin^2\theta = r.$

marginal : $q_\theta(\theta) = \dfrac{1}{2\pi},\quad q_r(r) = e^{-r^2/2}\, r,$

$\Rightarrow R, \theta$ **are independent.**

From $q_r(r)$, **one obtains the density of** $\mathbf{Z} := \mathbf{R}^2$,

which is $\dfrac{1}{2} e^{-z/2}, z > 0.$ (exponential distribution)

Note:

- Let Θ be uniform over $(0, 2\pi)$, and let R^2 be the exponential variable with mean 2. R, Θ are independent. Then the pair (X, Y) with

$$\begin{cases} X = R\cos\Theta, \\ Y = R\sin\Theta, \end{cases}$$

is standard 2-dim normal.

4. Normal Variates

- **An alternative method: inverse transformation**

  Approximating the inverse normal

  $$X = \Phi^{-1}(U)$$

  where $\Phi(x)$ is the standard normal CDF (analytical inverse is not available).

  A widely used method is Moro's algorithm.

  The inverse normal is approximated in a similar way to the implementation of cos, sin, log.
- It is also a more flexible approach because we'll need it later for stratified sampling and QMC.

4. Normal Variates

- **Another simple method:** Central Limit Theorem

Sample n copyies of U(0,1) r.v. $U_1, \ldots, U_n$ and return

$$X = \frac{\sum_{i=1}^{n} U_i - \frac{n}{2}}{\sqrt{n/12}}.$$

By CLT, X is approximately standard normal N(0,1). Often n is taken to be 12 to avoid the square root and division :

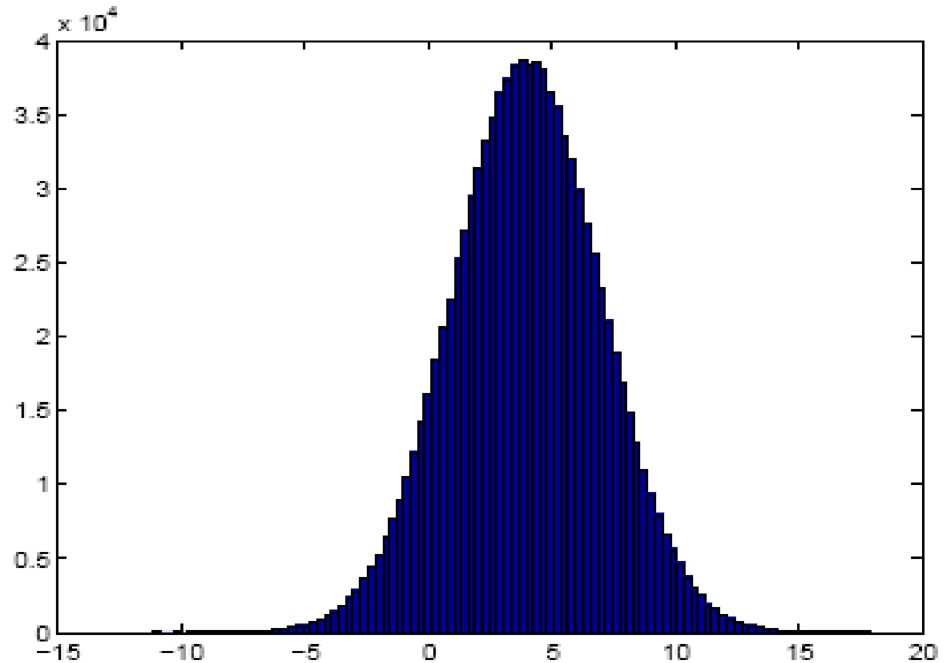
$$X = \sum_{i=1}^{12} U_i - 6$$

4. Normal Variates

In MATLAB, drawing from the standard normal distribution is implemented via the function **randn**.

For example, the following MATLAB program draws 10^6 samples from N(4, 9) and plots the corresponding histogram.

```
X = randn(1,10^6);
Z = 4 + 3*X;
hist(Z,100)
```



5. Generating Multivariate Normals

- In many financial applications one has to generate variates according to a multivariate normal distribution $N(\mu, \Sigma)$, with density

$$f(x) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu) \right).$$

- **Theorem: (Linear Transformation property)**

$$\textbf{If } Z \sim N(0, I), \textbf{ and } X = \boldsymbol{\mu} + AZ,$$

$$\textbf{then } X \sim N(\boldsymbol{\mu}, AA^T).$$

5. Generating Multivariate Normals

- Thus the problem of sampling X from $N(\mu, \Sigma)$ reduces to **finding a matrix A** for which

$$\mathbf{AA}^{\mathrm{T}} = \Sigma.$$

- The matrix A is **not uniquely** defined. The simplest choice is the **Cholesky factor** for the covariance matrix, i.e., an lower triangular matrix such that

$$\Sigma = \mathbf{AA}^{\mathrm{T}}.$$

- **In Matlab:** Cholesky factor is obtained by

$$\mathbf{A = chol}\ (\textstyle\sum).$$

Other methods can be used (will be discussed later).

Note:

- If $AA^T = \Sigma$, then $BB^T = \Sigma,$ if and only if B can be written as B = AU for some orthogonal matrix U (UU^T = I).

Thus we can fix an initial matrix A such that $AA^T = \Sigma$, and then find orthogonal matrix U.

5. Generating Multivariate Normals

- **The algorithm:**

(1) **Generate n independent standard normal variates $Z_1, \dots, Z_n$;**

(2) **Find a decomposition matrix A, such that**

$$AA^T = \Sigma.$$

(3) **Return**

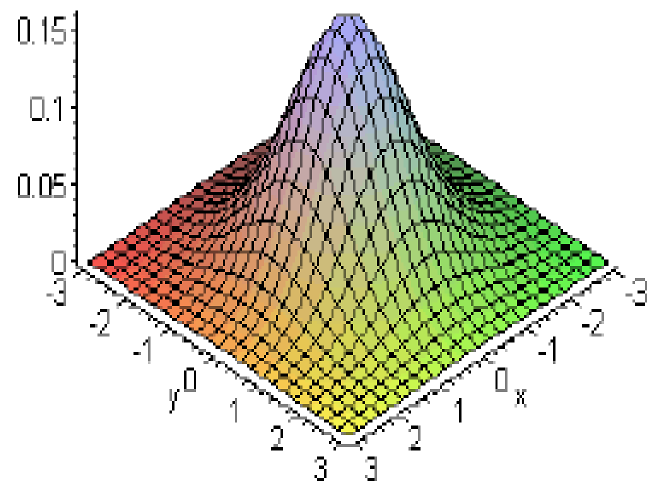
$$X = \mu + AZ, \quad Z = (Z_1, \dots, Z_n)^T.$$

Ex:

Let $X \sim N(\mu, \Sigma), \mu = (\mu_{1,} \mu_2)^T, \Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{pmatrix}.$

Cholesky factor of the variance matrix is

$$A = \begin{pmatrix} \sqrt{\sigma_{11}} & 0 \\ \dfrac{\sigma_{12}}{\sqrt{\sigma_{11}}} & \sqrt{\dfrac{\sigma_{11}\sigma_{22} - \sigma_{12}^2}{\sigma_{11}}} \end{pmatrix}.$$



Set

$$X = \mu + AZ, Z = (Z_1, Z_2)^T \sim N(0, I_2).$$

6. Generating Random Vector

- **Independent components:**

$$F(x_{1,} \ldots, x_n) = F_1(x_1) \cdots F_n(x_n),$$

Then we're back to the univariate case. We may sample each component individually — for example, via the inverse-transform method (or acceptance–rejection)

$$X_j = F_j^{-1}(U_j), \quad U_j \sim U(0,1)$$

or

$$F_j(X_j) = U_j, U_j \sim U(0,1).$$

6. Generating Random Vector

- **General case (with correlated components):**

  Writing the density function as

$$f(x_{1,}\cdots,x_n) = f_1(x_1)\, f_2(x_2 \mid x_1)\cdots f_n(x_n \mid x_1,\cdots,x_{n-1}).$$

- **The algorithm:**

  (1) generate X_1 from $f_1(x_1)$;

  (2) Given X_1, generate X_2 from f_2;

  …

  (n) Given X_1, …, $X_{(n-1)}$,

  generate X_n from f_n.

- The applicability of this approach depends on the knowledge of the **conditional distributions**.

Remarks

- Another, usually simpler, approach is to generate the random vector X by multidimensional AR.
- For high-dimensional distributions, efficient exact random variable generation is often difficult to achieve, and approximate generation methods are used instead --- Markov chain Monte Carlo (MCMC).
- The main idea of MCMC is to generate a Markov chain whose limiting distribution is equal to the desired distribution.

The algorithm (Sequential Inversion)

(1) Generate n independent uniform random numbers $U_1, \ldots, U_n$,

(2) Solve the equations:

$$\begin{cases} F_1(X_1) = U_1, \\ F_2(X_2 \mid X_1) = U_2, \\ \cdots\cdots\cdots \\ F_n(X_n \mid X_{1,} \ldots, X_{n-1}) = U_n. \end{cases}$$

and return $(X_1, \ldots, X_n)$.

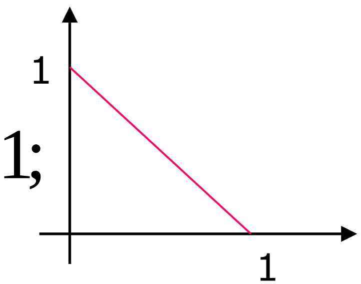
Note: There are n! admissible systems.

Example:

Consider random vector (X,Y) with density

$$f(x,y)=\begin{cases}6x, & x+y<1,\ x>0, y>0,\\ 0, & \text{otherwise}.\end{cases}$$

Method 1:



$$f_X(x)=\int_0^{1-x} f(x,y)dy=6x(1-x),\ \textbf{for } 0<x<1;$$

$$f_{Y|X}(y\,|\,x)=f(x,y)/f_X(x)=\frac{1}{1-x},\ 0<y<1-x.$$

$$F_X(x)=\int_0^x f_X(x)dx=3x^2-2x^3,\ 0<x<1;$$

$$F_{Y|X}(y\,|\,x)=\int_0^y f_{Y|X}(y\,|\,x)dy=\frac{y}{1-x},\ 0<y<1-x.$$

The algorithm is as follows

(1) **Generate two independent U_1 and U_2 from U(0,1);**

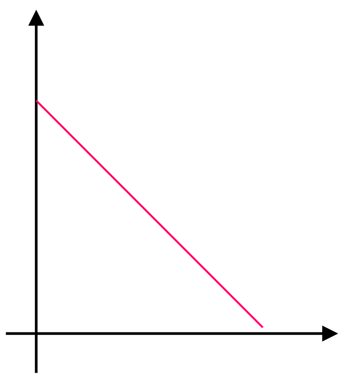
(2) **Solve the equations**

$$(*)\quad \begin{cases} 3X^2 - 2X^3 = U_1 \\ \dfrac{Y}{1-X} = U_2. \end{cases},$$

(3) **Return (X,Y).**

Note: The System (*) is difficult to solve.

Method 2: $f_Y(y)=\int_0^{1-y} f(x,y)dx=3(1-y)^2$, **for** $0<y<1$;



$$f_{X|Y}(x\mid y)=f(x,y)/f_Y(x)=\frac{2x}{(1-y)^2},0<x<1-y.$$

$$F_Y(y)=\int_0^{y} f_Y(y)dy=1-(1-y)^3,\ 0<y<1;$$

$$F_{X|Y}(x\mid y)=\int_0^{x} f_{X|Y}(x\mid y)dx=\frac{x^2}{(1-y)^2},0<x<1-y.$$

- **The algorithm is as follows:**

(1) Generate two independent U1 and U2 from U(0,1);

(2) Solve the equations
$$\begin{cases}1-(1-Y)^3=U_1\\ \dfrac{X^2}{(1-Y)^2}=U_2.\end{cases},$$

(3) Return (X,Y).

7. Method of Transformation

Goal: Generate samples from $F(x)$.

- Try to find a transformation $x=h(y)$, such that
  - The distribution function of $X= h(Y)$ is $F(x)$ (the RV Y have density $q(y)$).
  - Drawing samples of Y is easy.

Then we can

(1) Draw samples of Y from $q(y)$,

(2) Set $X = h(Y)$.

Note:

- The inverse transform method is a special form of transformation method.
- The pdf of X and Y are related by the next theorem.

Theorem:

Let X ~ f(x), a<x<b. Let Y = g(X), and suppose that g(x) is strictly increasing. Let X=h(Y) be the inverse function of Y=g(X). Then the density function of Y is

$$p(y) = \begin{cases} f(h(y))\,|h'(y)|, & \textbf{for } \boldsymbol{\alpha} < y < \boldsymbol{\beta}, \\ 0, & \textbf{othersise}. \end{cases}$$

where $\alpha = g(a), \beta = g(b).$

The similar is true when g(x) is strictly decreasing.

Note:

The result can be generalized to high dimension.

Examples

- **Ex 1:** (Uniform distribution)

  Suppose X~ U(a, b). X can be generated as

  $$X = a + (b-a)\,U,\ U \sim U(0,1).$$

- **Ex 2:** (Normal distribution)

  Suppose X~ N(a, b^2) . The X can be generated as

  $$X = a + b\,Z,\ Z \sim N(0,1).$$

- **Ex 3:** (Lognormal distribution)

  Suppose X = exp (Y), Y ~ N(a, b^2) . The X can be generated as

  $$X = \exp(a + b\,Z),\ Z \sim N(0,1).$$

8. Ratio-of-Uniform Method

- **Theorem:**

Let h(x) be a given **non-negative** function. Let

$$C = \left\{(u, v) : 0 < u \le \sqrt{h(v/u)}\right\}.$$

Let the point (U, V) be uniformly distributed over the region C. Define X = V/U. Then the pdf of X is

$$h(x) / \int_{-\infty}^{\infty} h(x)dx.$$

Note:

The method can be used to a density which is known up to a normalizing constant, i.e., the density has the form c h(x), where h(x) is known, but c can be unknown or difficult to compute.

8. Ratio-of-Uniform Method

- **The algorithm**

(1) Generate (U, V) uniformly over the set

$$C = \left\{ (u, v) : 0 < u \leq \sqrt{h(v/u)} \right\}.$$

(2) Return X = V/U.

$$\mathbf{X} \sim h(x) / \int_{-\infty}^{\infty} h(x) dx.$$

Proof (outline)

The joint density of (U, V) is

$$f_{U,V}(u,v) = 1/A, \quad \text{where} \quad A = \iint_C du dv.$$

Now put X=V/U. Then the joint density of (U, X) is

$$f_{U,X}(u,x) = u/A, \ (u,x) \in \left\{(u,x) : 0 < u \le \sqrt{h(x)}\right\}.$$

It follows that the marginal density of X is

$$f_X(x) = \int_0^{\sqrt{h(x)}} \frac{u}{A} du = \frac{h(x)}{2A}.$$

Since $f_X(x)$ is a density, we must have $1 = \dfrac{\int_{-\infty}^{\infty} h(x)}{2A} dx.$

Thus $f_X(x) = \dfrac{h(x)}{\int_{-\infty}^{\infty} h(x) dx}.$

Example:

Consider the Cauchy distributi on with density

$$\mathbf{f(x) = \frac{1}{\pi(1 + x^2)}}.$$

Let $\mathbf{h(x) = 1/(1 + x^2)}$**.**

What is the region C defined by

$$C = \left\{(u,v) : 0 < u \le \sqrt{h(v/u)}\right\}?$$

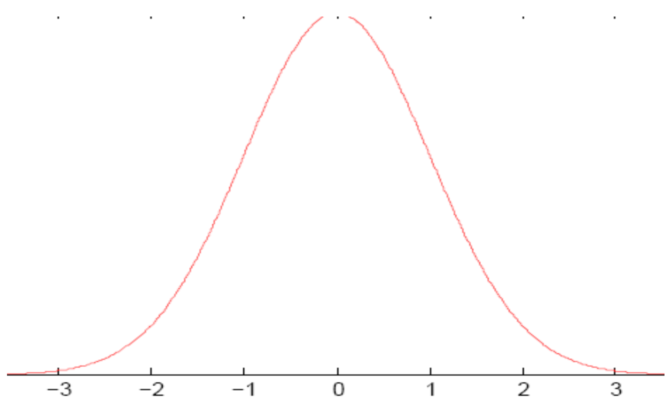
We have

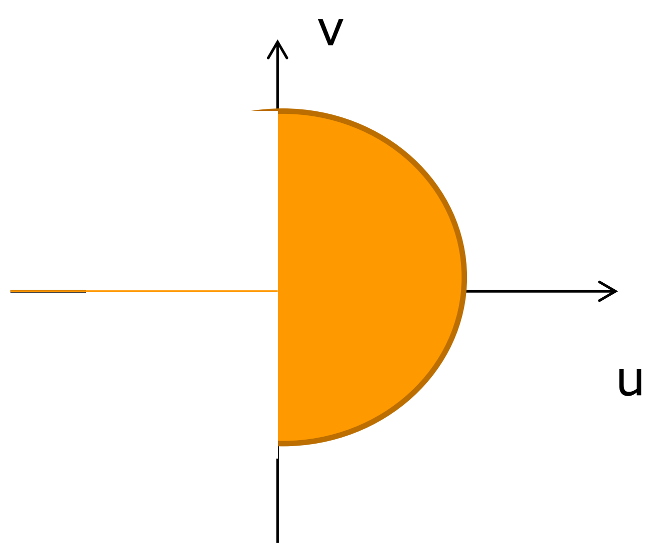
$$0 < u \le \sqrt{h(v/u)} \iff \begin{cases} u^2 \le h(v/u) \\ u > 0 \end{cases} \iff \begin{cases} u^2 + v^2 \le 1 \\ u > 0 \end{cases}$$

Example

The Cauchy distribution can be generated as

1. Generate $U \sim U(0,1)$, $V \sim U(-1,1)$;
2. If $U^2 + V^2 \leq 1$, return $X := V/U$.





Remarks:

- This theorem leads to a method of generating RV with the density proportional to $h(x)$.
- The only technical problem is how to generate points uniformly over the region C.
- Usually, this can be done by enclosing C with a rectangle R with sides parallel to u and v axes.
- Generating points uniformly within R is easy.
  If such a point also falls in C, it is accepted and the ratio V/U is accepted, otherwise it is rejected.

Theorem:

Let $b = \sup_x \sqrt{h(x)},\ c = \inf_x\ x\sqrt{h(x)},\ d = \sup_x\ x\sqrt{h(x)}.$

and $R = \{(u,v) : 0 \le u \le b, c \le v \le d\}.$

Then the rectangle R encloses C.

Proof. Suppose $(u,v) \in C.$ Put x=v/u. Then

$$0 < u \le \sqrt{h(x)} \le b.$$

Now suppose v>0, then x>0. Then

$$v = xu \le x\sqrt{h(x)} \le d.$$

If v<=0, then x<=0, and $v = xu \ge x\sqrt{h(x)} \ge c.$

The Algorithm:

均匀分布

(1) Generate two independent random numbers U_1, V_1;

(2) Set $U = bU_1$, $V = c+(d-c)V_1$;

(this generates uniformly distributed point in R)

(3) Set $X = V/U$;

(4) If $U^2 \le h(X)$ (implying the point (U, V) falls in C),

then take X as the desired realization,

Otherwise, go to step (1).

* We must at first calculate the values of b, c, d.

Ex: A ratio-of uniforms method for N(0,1)

We take $\boldsymbol{h}(\boldsymbol{x}) = \exp\left(-\boldsymbol{x}^2/2\right)$. Therefore

$$b = \sup_x \sqrt{h(x)} = 1,\ c = \inf_x\ x\sqrt{h(x)} = -\sqrt{2/e},$$

$$d = \sup_x\ x\sqrt{h(x)} = \sqrt{2/e}.$$

$$C = \left\{(u,v) : 0 < u \le \sqrt{h(v/u)}\right\},\ R = \left\{(u,v) : 0 \le u \le b, c \le v \le d\right\}.$$

The algorithm above can be used directly.

The acceptance probability is

$$P = \frac{\text{measure}\,(C)}{\text{measure}\,(R)} = \frac{\frac{1}{2}\int_{-\infty}^{\infty} h(x)dx}{2\sqrt{2/e}} = \frac{\sqrt{e\pi}}{4} \approx 0.731.$$

9. Simulating Copulas

- **Why do we need copulas?**

- The modeling of dependence structures (or copulas) is undoubtedly one of the key challenges for modern financial engineering.

- First applied to credit-risk modeling, copulas are now widely used across a range of derivatives transactions, asset pricing techniques, and risk models, and are a core part of the financial engineer's toolkit.

- However, copulas are complex and their applications are often misunderstood. Incorrectly applied, copulas can be hugely detrimental to a model or algorithm.

9. Simulating Copulas

- Copulas provide a way to create distributions to model correlated multivariate data.
- Using a copula, we can construct a multivariate distribution by:
  - specifying marginal univariate distributions, and
  - choosing a particular copula to provide a correlation structure between variables.

9. Simulating Copulas

- One step in MC simulation is choice of probability distributions for random inputs.
- Selecting a distribution for each individual variable is straightforward, but deciding what dependencies should exist between the inputs may not be.
- There may be little or no information on which to base any dependence in simulation.
- It is a good idea to experiment with different possibilities, in order to determine the model's sensitivity.

9. Simulating Copulas

- Traditional equity risk models focus on estimating stock return variance-covariance matrix. Ignoring high-order moments, they implicitly assumes normal return distributions.
- The normality assumption is insufficient in risk management. Moving away from normality requires a tractable technique to allow investigation of alternative distributions.
- Copula is a good choice since it enriches our distribution selection menu. Copulas have primary and direct applications in the simulation of dependent variables.

9. Simulating Copulas

- It can be difficult to actually generate random inputs with dependence when they have distributions that are not from a standard multivariate distribution.
- Further, some standard multivariate distributions can model only very limited types of dependence.
- Simulation of financial risk may have random inputs that represent different sources of insurance losses. These inputs might be modeled as lognormal RVs.
- A reasonable question is how dependence between these two inputs affects results of the simulation.

9. Simulating Copulas

- Apart from the family of normal distributions, there do not seem to be other popular families of distributions which allow a natural multivariate generalization such that one can easily simulate dependent RVs. (some)
- Often the joint distribution of RVs can only be explicitly computed if the RVs are independent.
- The concept of copulas is a very useful tool to overcome this problem.

Some we can do

- multivariate normal
- multivariate t
- multinomial (multivariate binomial)
- Dirichlet (multivariate beta)
- multivariate exponential

- Can we just put "multivariate" in front of any distribution name? Sort of: but it won't be unique.
- For dim > 1, we more often force our problem into a list of distributions we can do.

9. Simulating Copulas

- What are copulas?
- How to use copulas to generate data from multivariate distributions?
   - When there are complicated relationships among the variables, or
   - When the individual variables are from different distributions?

Introducing Copulas

- The history of copulas began with Fréchet (1950). Consider problem: given the distribution functions F_1 and F_2 of two RVs X_1 and X_2 defined $(\Omega, \mathcal{F}, \mathrm{P})$, what can be said about the set $\Gamma(F_1, F_2)$ of the bivariate d.f.'s whose marginals are F_1 and F_2?

- The set $\Gamma(F_1, F_2)$ is not empty since, if X_1 and X_2 are independent, then the distribution function

$$(x_1, x_2) \to F(x_1, x_2) = F_1(x_1)F_2(x_2)$$

  always belongs to $\Gamma(F_1, F_2)$.

- But, it was not clear which the other elements of $\Gamma(F_1, F_2)$ were.

Introducing Copulas

The CDF of a d-dim RV $\mathbf{X} := (\mathbf{X_1}, \cdots, \mathbf{X_d})$

$$\mathbf{F}(x_1,\ldots,x_d) = \mathbf{P}(\mathbf{X_1} \le x_1, \cdots, \mathbf{X_d} \le x_d).$$

Margin: $\mathbf{F_i}(x_i) = \mathbf{F}(\infty,\ldots,\infty, x_i, \infty, \cdots, \infty).$

It is not enough to know the margins $\mathbf{F_1},\ldots,\mathbf{F_d}$ in order to determine F. Addtiona lly, it is required to know how the marginal laws are coupled.

This is achieved by means of a copula of $\mathbf{X} = (\mathbf{X_1}, \cdots, \mathbf{X_d})$.

Knowing the margins and the copula is equivalent to knowing the distributi on.

Copulas: The Basic Idea

Consider a RV $(X_1,\ldots,X_d)$ of dependent components $X_i \sim F_i(x)$. Set

$$(U_1,\ldots,U_d) := (F_1(X_1),\ldots,F_d(X_d)).$$

It has uniform margins.

The copula of $(X_1,\ldots,X_d)$ is defined as the joint CDF of $(U_1,\ldots,U_d)$:

$$C(u_1,\ldots,u_d) = P(U_1 \le u_1,\ldots,U_d \le u_d).$$

Copulas: The Definition

A function $C:[0,1]^d$ is called a d-dim copula,

if there is a probability space (Ω, F, P) supporting

RV $(U_1,\ldots,U_d)$, such that

$$U_k \sim U(0,1) \quad \text{for all } k = 1,\ldots,d,$$

and

$$C(u_1,\ldots,u_d) = P(U_1 \le u_1,\ldots,U_d \le u_d).$$

Copulas: The Definition

A copula C is defined as the CDF on $[0,1]^d$ of a RV $(U_1,\ldots,U_d)$, where each U_i has uniformly distribute d marginals, i.e.,

$$C(1,\ldots,1,x_i,1,\ldots,1) = x_i, \quad \text{for } i \in \{1,\ldots,d\}.$$

Basic examples of copulas

- **Independence copula:**

  $\Pi(\mathbf{u}) = u_1 \cdots u_d$

  It is associated with RV

  $$\mathbf{U} = (U_1, \ldots, U_d)$$

  whose components are independent and uniformly distributed on [0,1].

Basic examples of copulas

- **The comonotonicity copula:**

  $M(\mathbf{u}) = \min\{u_1,\ldots,u_d\}$

  It is associated with a vector

  $\mathbf{U} = (U_1,\ldots,U_d)$

  of RVs uniformly distributed on [0,1] and such that $U_1 = \cdots = U_d$ almost surely.

  Since $\mathbf{U_1 = \cdots = U_d =: U}$, **we have**

  $$\mathbf{P(U_1 \le u_1, \cdots U_d \le u_d) = P(U \le \min(u_1,\ldots,u_d))}$$

  $$\mathbf{= \min(u_1,\ldots,u_d).}$$

Basic examples of copulas

- **The counter-monotonicity copula:**

  $W_2(u_1,u_2) = \max\{u_1 + u_2 - 1, 0\}$

  It is associated with a vector $\mathbf{U} = (U_1,U_2)$ of RVs uniformly distributed on [0,1] and such that $U_1 = 1-U_2$ almost surely.

$$\mathbf{P}(\mathbf{U}_1 \le \mathbf{u}_1, \mathbf{U}_2 \le \mathbf{u}_2) = \mathbf{P}(1 - \mathbf{U}_2 \le \mathbf{u}_1, \mathbf{U}_2 \le \mathbf{u}_2)$$

$$= \begin{cases} \mathbf{P}(1 - \mathbf{u}_1 \le \mathbf{U}_2 \le \mathbf{u}_2), & \mathbf{u}_2 + \mathbf{u}_1 - 1 \ge 0 \\ 0, & \textbf{otherwise.} \end{cases}$$

$$= \max(\mathbf{u}_1 + \mathbf{u}_2 - 1, 0).$$

➢ Note: It is not possible to have a counter-monotonicity copula with $d > 2$.

Remark:

- Copula contains all information on the dependence structure between the components of $(X_1, \ldots, X_d)$. The copula controls the joint distribution.
- Marginal contains all information on marginal distributions.
- The relationship between copulas and multivariate distributions is established by Sklar Theorem.
- Sklar Theorem argues that any given multivariate distribution function is expressible as copula of its marginals.

Theorem (Sklar)

Let $(X_1,\dots,X_d)$ be random vector with marginal F_i and joint CDF F. Then there exists a copula C, such that

$$F(x_1,\dots,x_d) = C(F_1(x_1),\dots,F_d(x_d)). \qquad (*)$$

If F_i are continuous, then C is unique.

F is a function of its marginals and the copula.

Conversely, if C is a d-dim copula and F_i are univariate functions, then the function F defined via (*) is a d-dim distribution function.

F $\Longleftrightarrow$ Fi + C

F<=>Fi+C

Understanding

Let $(X_1,\ldots,X_d)$ be RV with marginal F_i and joint distribution on F.

Let $U_i = F_i(X_i)$, then $U_i \sim U([0,1])$. Hence, the distribution on function of $(U_1,\ldots,U_d)$ is a copula by definition :

$$C(u_1,\ldots,u_d) = P(U_1 \le u_1,\ldots,U_d \le u_d).$$

Then

$$\begin{aligned} F(x_1,\ldots,x_d) &= P(X_1 \le x_1,\ldots,X_d \le x_d) \\ &= P(F_1^{-1}(U_1) \le x_1,\ldots,F_d^{-1}(U_d) \le x_d) \\ &= P(U_1 \le F_1(x_1),\ldots,U_d \le F_d(x_d)) \\ &= C(F_1(x_1),\ldots,F_d(x_d)). \end{aligned}$$

(unique?)

CDF to copula

Understanding

Let C is a copula and let (Ω, F, P) is a probability space supporting a RV $(U_1, \ldots, U_d) \sim C$.

Define RV $(X_1, \ldots, X_d)$ by $X_i = F_i^{-1}(U_i), i = 1, \ldots, d$.

Then $X_i \sim F_i$. Furthermore,

$$P(X_1 \le x_1, \ldots, X_d \le x_d) = P(F_1^{-1}(U_1) \le x_1, \ldots, F_d^{-1}(U_d) \le x_d)$$

F(x1, …, Xxd

$$= P(U_1 \le F_1(x_1), \ldots, U_d \le F_d(x_d))$$

$$= C(F_1(x_1), \ldots, F_d(x_d)).$$

copula定义

$\Rightarrow C(F_1(x_1), \ldots, F_d(x_d))$ is the CDF of $(X_1, \ldots, X_d)$.

Copula to CDF

Remarks

Sklar's theorem has a clear message:

- Marginal distributions and dependence structure of n-dim random vector can be strictly separated.
- While marginal distributions are determined by univariate distribution functions, dependence structure is determined by the copula.
- Any multivariate distribution can be split into its univariate margins and a copula.
- Conversely, combining some given margins with a given copula, one can build multivariate distribution.

Remark:

- The theorem allows us to divide the treatment of multivariate distribution into two often easier subtreatments:

- Investigation of the univariate marginal laws;
- Investigation of a copula (standardized and often more convenient multivariate distribution function).

- The marginals are the easy part. The copula is the hard part.

Theorem

For a given copula C and marginal cdfs $\{F_i\}$, define

$$F(x_1,\ldots,x_d) = C(F_1(x_1),\ldots,F_d(x_d)).$$

If $(U_1,\ldots,U_d)$ has cdf $C(u_1,\ldots,u_d)$, then the RV

$$X = (X_1,\ldots,X_d) = (F_1^{-1}(U_1),\ldots,F_d^{-1}(U_d))$$

has joint cdf F and marginals $\{F_i\}$.

This theorem provides an algorithm of simulating from F: If a multivariate distribution is expressed by a copula C and the marginal functions, one can simulate the RV as described below.

Simulating Multivariate Distribution via Copula

- An algorithm to generate $(X_1, \ldots, X_d)$ having known CDF F.

Let C be the copula having the property

$$F(x_1,\ldots,x_d) = C(F_1(x_1),\ldots,F_d(x_d)).$$

Algorithm : Copula-marginal sampling

(1) Generate $(U_1,\ldots,U_d) \sim C(u_1,\ldots,u_d)$;

(2) Return $X = (F_1^{-1}(U_1),\ldots,F_d^{-1}(U_d))$.

- Any copula we like with any margins we like.

Remarks

- Sampling from copula is not necessarily any easier than sampling from other multivariate distribution.
- The marginal distributions are defined through d one-dim curves, but the copula is inherently d-dimensinal.
- The algorithm above is just pushes the difficulty of sampling from multivariate distribution into the copula.

Example

- Indendence copula :

Suppose $U_1,\ldots,U_d \sim U(0,1)$ are independent.

The corresponding copula is given by

$$C(u_1,\ldots,u_1) = P(U_1 \le u_1,\ldots,U_d \le u_d)$$

$$= \prod_{i=1}^{d} P(U_i \le u_i) = u_1 \ldots u_d.$$

So $X_j = F_j^{-1}(U_j)$ are independent.

This provides a method for simulating RV X with independent component.

Example

- Gaussian copula, as the name suggests, is the copula embedded in the multivariate Gaussian distribution.
- It represents the dependence structure between Gaussian variables.
- The bivariate Gaussian copula takes the following form: (next page)

$$F(x_1, \ldots, x_d) = C(F_1(x_1), \ldots, F_d(x_d)).$$

U1, … , ud

Example

Gaussian copula :

$$C(u_1,\ldots,u_d) = \Phi_\Sigma(\Phi^{-1}(u_1),\ldots,\Phi^{-1}(u_d)),\ u_i \in [0,1],$$

where $\Phi(.)$ is the cdf of $N(0,1)$, and Φ_Σ is the CDF of $N(0,\Sigma)$, where Σ is a positive-definite matrix.

For $d = 2, \Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$, then

$$C(u_1, u_2) = \int_{-\infty}^{\Phi^{-1}(u_1)} \int_{-\infty}^{\Phi^{-1}(u_2)} \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left(-\frac{s^2 - 2\rho s t + t^2}{2(1-\rho^2)}\right) ds\,dt.$$

Example

Simulating Gaussian copula :

$$C(u_1,\dots,u_d) = \Phi_\Sigma(\Phi^{-1}(u_1),\dots,\Phi^{-1}(u_d)),\ u_i \in [0,1].$$

1. Perform Cholesky decomposition of Σ: $\Sigma = AA^T$.

2. Generate $Z = (Z_1,\dots,Z_d)^T \sim N(0, I_d)$.

3. Set $X = AZ$;

4. Set $U = (\Phi(X_1),\dots,\Phi(X_d))^T$.

$$\text{Note}: C(u_1,\dots) = P(U_1 \le u_1,\dots) = P(\Phi(\Phi_1) \le u_1,\dots)$$

$$= P(X_1 \le \Phi^{-1}(u_1)) = \Phi_\Sigma(\Phi^{-1}(u_1),\dots,).$$

Remarks

- Copulas are general tool to describe dependence structures. They allow us to separate the problem to specify marginal distributions and model the dependence.
- Various copulas have their form mainly because of mathematical tractability. Their applicability should be verified in each case.
- If the selected copula is wrong and does not fit the reality, risk measure and prices calculated from the model can be highly misleading.
- **Which copula to use?**

10. Sampling from Specific Distributions

- **Devroye, Non-Uniform Random Variate generation, 1986.**

- **Note:**

  If the distribution has a name (normal, Possion, Gamma, beta, etc.), it is probably already in Matlab or R, or …

Exercise 1 (dead line: 17 March)

1. Suppose we want to sample from the density

$$f(x) = x + 1/2, \quad 0 < x < 1.$$

(1) **Using the inverse transform methods, simulate 1000 values from** f**;**

(2) Using the acceptance-rejection method, simulate another 1000 values from f**.**

(3) Which algorithm is more efficient?

Exercise 1 (dead line: 17 March)

2. Suppose we want to simulate $|Z|$, where $Z \sim N(0,1)$.

   The pdf of $|Z|$ is

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

Take $g(x) = e^{-x}, \quad 0 < x < +\infty.$

(1) Determine the value of c such that $c = \max \dfrac{f(x)}{g(x)}.$

(2) Using acceptance-rejection algorithm to simulate 1000 values of $|Z|$.

(3) How to recover Z from the simulated values of $|Z|$?

Exercise 1 (dead line: 26 March)

3. Suppose that

$$F(x) = \prod_{j=1}^{K} F_j(x),$$

where $F_j(\cdot)$ are CDFs from which we can sample easily (x is univariate).

Describe a way of sampling $X \sim F(x)$.

Exercise 1 (dead line: 17 March)

4. Suppose for some real numbers $a < b$, and some pdf $f(x)$ with associated CDF $F(x)$, $-\infty < x < \infty$, we want to generate random variates having the truncated pdf

$$g(x) = \begin{cases} \dfrac{f(x)}{F(b) - F(a)}, & a \le x \le b, \\ 0, & \text{else.} \end{cases}$$

Assume the inverse CDF $F^{-1}(\cdot)$ can be computed. Explain how to generate variates from the above truncated pdf.

Exercise 1 (dead line: 17 March)

5. For the beta pdf

$$f(x) = 12x^2(1-x), 0 \le x \le 1$$

implement the acceptance-rejection approach, and for a sample of 100,000 beta variates, compute the average number of uniform variates required to output one beta variate.

Projects:

- Review the generation of random samples from standard statistical distributions.
- Generalize the Acceptance-Rejection method to high dimensions, give examples.
- Review the copula methods and applications in finance and insurance.

The End of Chapter 3