

Método Polar

Generación de eventos en Procesos de Poisson

Patricia Kisbye

FaMAF

24 de abril, 2008

Métodos de generación de v. a. normales

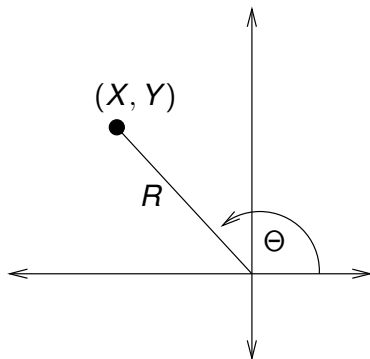
- ▶ Notar: Si $X \sim N(0, 1)$, entonces $\mu + \sigma X \sim N(\mu, \sigma^2)$, por lo cual es suficiente saber generar $X \sim N(0, 1)$.
- ▶ Método de Box - Muller (1958) - Se generan dos variables normales independientes X e Y .
- ▶ Estrategia: utilizar una de las dos normales como respuesta en las llamadas pares.
- ▶ Transformaciones de Box - Muller: utilizan funciones trigonométricas.
- ▶ Se recomienda utilizar una combinación de generadores. Los generadores de congruencias producen puntos espiralados.
- ▶ Método Polar: Marsaglia and Bray (1964) - Eliminan los cálculos trigonométricos. Reducción del tiempo de ejecución en un 27%.

Método de Box Muller

Si X e Y son normales estándar independientes, entonces

$$R^2 = X^2 + Y^2, \quad \tan(\theta) = \frac{Y}{X}$$

determinan variables R^2 y Θ independientes.



- ▶ $f_X(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$
- ▶ $f_Y(y) = \frac{1}{\sqrt{2\pi}} e^{-y^2/2}$
- ▶ $f_{X,Y}(x,y) = \frac{1}{2\pi} e^{-(x^2+y^2)/2}$
- ▶ $R^2 = X^2 + Y^2$
- ▶ $\Theta = \arctan(\frac{Y}{X})$

Método de Box-Muller

$$f_{R^2, \Theta}(d, \theta) = \frac{1}{J} \frac{1}{2\pi} e^{-d/2} = \frac{1}{2} \frac{1}{2\pi} e^{-d/2},$$

donde

$$J = \det \begin{pmatrix} \frac{\partial d}{\partial x} & \frac{\partial d}{\partial y} \\ \frac{\partial \theta}{\partial x} & \frac{\partial \theta}{\partial y} \end{pmatrix}, \quad d = x^2 + y^2, \quad \theta = \arctan(y/x)$$

$$R^2 \sim \mathcal{E}(\frac{1}{2}) \text{ y } \Theta \sim \mathcal{U}(0, 2\pi).$$

$$X = R \cos(\theta), \quad Y = R \sin(\theta)$$

Método de Box-Muller

Generar $U_1, U_2 \sim \mathcal{U}(0, 1)$;

$R^2 \leftarrow -2 \log(U_1)$;

$\Theta \leftarrow 2\pi U_2$;

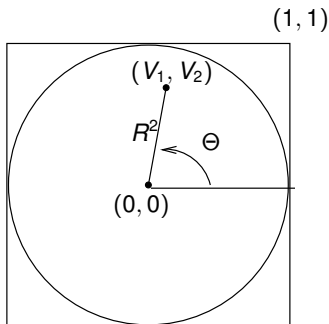
$X \leftarrow R \cos(\Theta)$;

$Y \leftarrow R \sin(\Theta)$

Método de Box-Muller

Transformaciones de Box-Muller:

- ▶ $X \leftarrow \sqrt{-2 \log(U_1)} \cos(2\pi U_2)$
- ▶ $Y \leftarrow \sqrt{-2 \log(U_1)} \sin(2\pi U_2)$
- ▶ El cálculo de funciones trigonométricas lo hace poco eficiente.
- ▶ Idea de Marsaglia: Utilizar v. a. uniformes V_1 y V_2 en el círculo unitario y utilizar nuevamente coordenadas polares.



Método polar

- ▶ Generar un par (V_1, V_2) uniformemente distribuido en el cuadrado.
- ▶ $C =$ círculo unitario,

$$P(A \mid A \subset C) = \int \int_C \frac{1}{\pi} \mathbb{I}_{A \cap C}(x, y) \, dx \, dy$$

- ▶ Si $(V_1, V_2) \in C$, entonces resulta uniformemente distribuido en el círculo unitario.
- ▶ $R^2 = V_1^2 + V_2^2$ resulta uniforme en $(0, 1)$.

$$P(R^2 < a) = \frac{1}{\pi} \pi \cdot a = a.$$

- ▶ $\Theta = \arctan V_2/V_1$ es uniforme en $(0, 2\pi)$:

$$P(\Theta < \alpha) = \frac{\alpha}{2\pi}.$$

$$\sin \Theta = \frac{V_2}{R} = \frac{V_2}{\sqrt{V_1^2 + V_2^2}} \quad \cos \Theta = \frac{V_1}{R} = \frac{V_1}{\sqrt{V_1^2 + V_2^2}}$$

Transformaciones de Box Muller resultan:

$$X = (-2 \log U)^{1/2} \frac{V_1}{(V_1^2 + V_2^2)^{1/2}}$$

$$Y = (-2 \log U)^{1/2} \frac{V_2}{(V_1^2 + V_2^2)^{1/2}}$$

- R^2 y Θ son independientes, luego podemos tomar

$$U = R^2 = V_1^2 + V_2^2$$

Método polar

Tomamos $S = R^2$, y obtenemos

$$X = (-2 \log S)^{1/2} \frac{V_1}{S^{1/2}}$$

$$Y = (-2 \log S)^{1/2} \frac{V_2}{S^{1/2}}$$

Método polar (última versión)

repeat

Generar $V_1, V_2 \sim \mathcal{U}(-1, 1)$;

$S \leftarrow V_1^2 + V_2^2$

until $S < 1$;

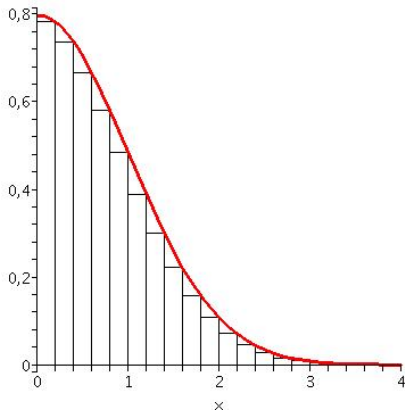
$X \leftarrow \sqrt{\frac{-2 \log S}{S}} V_1$;

$Y \leftarrow \sqrt{\frac{-2 \log S}{S}} V_2$

- ▶ Requiere $\frac{4}{\pi} \approx 1.273$ iteraciones.
- ▶ Entre un 9% y 31% más rápido en FORTRAN.

Generación de v. a. normales

Existen otros métodos de generación de v. a. normales. Uno de ellos es el método *rectangle-wedge-tail*:



$|Z|, Z \sim N(0, 1)$

- ▶ 15 rectángulos ($\approx 92\%$)
- ▶ 15 "triángulos".
- ▶ 1 cola.

Método de composición:

- ▶ 15 distr. uniformes.
- ▶ 16 restantes: método de rechazo.

Referencia: Donald E. Knuth,
Seminumerical Algorithms, p.p.
122 en adelante.

Generación de un Proceso de Poisson

En un proceso de Poisson homogéneo de razón λ , los tiempos de llegada entre eventos sucesivos son exponenciales con razón λ (media $1/\lambda$).

Para generar los primeros n eventos de un Proceso de Poisson homogéneo, generamos exponenciales $X_i \sim \mathcal{E}(\lambda)$, $1 \leq i \leq n$:

- ▶ Primer evento: al tiempo X_1 .
- ▶ j -ésimo evento: al tiempo $X_1 + X_2 + \dots + X_j$, para $1 \leq j \leq n$.

Para generar los eventos en las primeras T unidades de tiempo, generamos eventos hasta que $j + 1$ excede a T .

Proceso de Poisson homogéneo

Generación de eventos en el intervalo $[0, T]$

$t \leftarrow 0, I \leftarrow 0;$

while true do

 Generar $U \sim \mathcal{U}(0, 1);$

if $t - \frac{1}{\lambda} \log U > T$ **then**

stop

else

$t \leftarrow t - \frac{1}{\lambda} \log U;$

$I \leftarrow I + 1;$

$S[I] \leftarrow t$

end

end

- ▶ I : número de eventos.
- ▶ $S[I]$: tiempo en el que ocurre el evento I .
- ▶ $S[1], S[2], \dots$, están en orden creciente.

Proceso de Poisson homogéneo

Otro método para simular los primeros eventos se fundamenta en lo siguiente: sea $N(T)$: el número de eventos hasta el tiempo T :

Dado $N(T)$, la distribución de los tiempos de arribo en un Proceso de Poisson homogéneo de razón λ es uniforme en $(0, T)$.

Luego, para generar los eventos hasta el tiempo T :

- ▶ Generar una v. a. Poisson de media λT , y tomar $n = N(T)$.
- ▶ Generar n v. a. uniformes $U_1, U_2, \dots, U_n$.
- ▶ Los tiempos de arribo son: $TU_1, TU_2, \dots, TU_n$.
- ▶ ¿Es más eficiente? Notar que hay que ordenar los tiempos de arribo.

Proceso de Poisson no homogéneo

En un Proceso de Poisson no homogéneo, con intensidad $\lambda(t)$:

- ▶ los incrementos no son estacionarios, es decir, no dependen de la longitud del intervalo,
- ▶ la intensidad $\lambda(t)$ indica la tasa de ocurrencia de los eventos,
- ▶ si $\lambda(t) \leq \lambda$, entonces $\lambda(t) = \lambda \cdot \frac{\lambda(t)}{\lambda}$ con $\frac{\lambda(t)}{\lambda} < 1$,
- ▶ podemos considerar $p(t) = \lambda(t)/\lambda$ como la probabilidad de ocurrencia de un evento en un Proceso de Poisson homogéneo de razón λ .

Método de adelgazamiento o muestreo aleatorio

Proceso de Poisson no homogéneo

Generación de eventos en el intervalo $[0, T]$

$t \leftarrow 0, I \leftarrow 0;$

while true do

Generar $U \sim \mathcal{U}(0, 1);$

if $t - \frac{1}{\lambda} \log U > T$ **then**

stop

else

$t \leftarrow t - \frac{1}{\lambda} \log U;$

if $U < \lambda(t)/\lambda$ **then**

$I \leftarrow I + 1;$

$S[I] \leftarrow t$

end

end

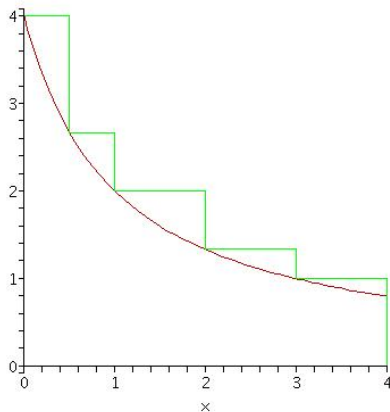
end

- ▶ I : número de eventos.
- ▶ $S[1], S[2], \dots$: tiempos de eventos.
- ▶ El algoritmo es más eficiente cuánto más cerca esté λ de $\lambda(t)$.

Proceso de Poisson no homogéneo

Una mejora consiste en particionar el intervalo $(0, T)$ en subintervalos, y aplicar el algoritmo anterior en cada uno de ellos:

$$0 = t_0 < t_1 < \dots < t_k < t_{k+1} = T$$



$$\lambda(s) \leq \lambda_j, \\ \text{si } j-1 \leq s < j$$

Proceso de Poisson no homogéneo

```
 $t \leftarrow 0; J \leftarrow 1; I \leftarrow 0;$   
while true do  
  Generar  $U \sim \mathcal{U}(0, 1);$   
   $X \leftarrow -\frac{1}{\lambda_J} \log U;$   
  while  $t + X > t_J$  do  
    if  $J = k + 1$  then  
      stop  
    end  
     $X \leftarrow (X - t_J + t) \frac{\lambda_J}{\lambda_{J+1}};$   
     $J \leftarrow J + 1$   
  end  
   $t \leftarrow t + X;$   
  Generar  $V \sim \mathcal{U}(0, 1);$   
  if  $V < \lambda(t)/\lambda_J$  then  
     $I \leftarrow I + 1;$   
     $S[I] \leftarrow t$   
  end  
end
```

Proceso de Poisson no homogéneo

Otra mejora sobre este algoritmo es considerar el mínimo de $\lambda(s)$ en cada subintervalo.

Por ejemplo, en un intervalo (t_{i-1}, t_i) :

$$\blacktriangleright \lambda_{min} = \min\{\lambda(s) \mid s \in (t_{i-1}, t_i)\}$$

En este intervalo los eventos ocurren con intensidad

$$(\lambda(s) - \lambda_{min}) + \lambda_{min},$$

- ▶ Generar eventos de un P. P. homogéneo con razón λ_{min} .
- ▶ Intercalarlos con eventos de un P. P. no homogéneo con intensidad

$$\tilde{\lambda}(s) = \lambda(s) - \lambda_{min}.$$

- ▶ Ventaja: El número promedio de uniformes a generar disminuye en $\lambda_{min}(t_i - t_{i-1})$.

Proceso de Poisson no homogéneo

Otro método consiste en generar directamente los sucesivos tiempos de evento.

- ▶ $S_1, S_2, \dots$: sucesivos tiempos de evento.
- ▶ Estos tiempos de evento no son independientes.
- ▶ Para generar S_{i+1} , utilizamos el valor generado S_i .
- ▶ Dado que ha ocurrido un evento en $t = S_i$, el tiempo hasta el próximo evento es una variable aleatoria con la siguiente distribución:

$$\begin{aligned}F_s(x) &= P\{\text{ocurra un evento en } (s, s+x)\} \\&= 1 - P\{0 \text{ eventos en } (s, s+x)\} \\&= 1 - \exp\left(\int_s^{s+x} \lambda(y) dy\right) = 1 - \exp\left(\int_0^x \lambda(s+y) dy\right)\end{aligned}$$

Proceso de Poisson no homogéneo

Ejemplo

Describir un algoritmo para la generación de eventos de un P. P. no homogéneo con función de intensidad

$$\lambda(t) = \frac{1}{t+3}, \quad t \geq 0.$$

$$\int_0^x \lambda(s+y) dy = \int_0^x \frac{1}{s+y+3} dy = \log \left(\frac{x+s+3}{s+3} \right).$$

$$F_s(x) = 1 - \frac{s+3}{x+s+3} = \frac{x}{x+s+3}.$$

$$u = F_s(x) \quad \Rightarrow \quad x = \frac{u(s+3)}{1-u}.$$

Proceso de Poisson no homogéneo

$$F_s^{-1}(u) = (s + 3) \frac{u}{1 - u}.$$

Los sucesivos tiempos de eventos son entonces:

$$S_1 = \frac{3U_1}{1 - U_1}$$

$$S_2 = S_1 + (S_1 + 3) \frac{U_2}{1 - U_2} = \frac{S_1 + 3U_2}{1 - U_2}$$

$$\vdots$$

$$S_j = S_{j-1} + (S_{j-1} + 3) \frac{U_{j-1}}{1 - U_{j-1}} = \frac{S_{j-1} + 3U_j}{1 - U_j}$$

Proceso de Poisson no homogéneo

Generación de eventos utilizando el método de la T. Inversa

$I = 0;$

$S[0] = 0;$

while true do

 Generar $U \sim \mathcal{U}(0, 1);$

if $(S[I] + 3U)/(1 - U) > T$ **then**

stop

else

$I \leftarrow I + 1;$

$S[I] \leftarrow (S[I - 1] + 3U)/(1 - U);$

end

end
