Generación de variables aleatorias normales Generación de eventos en Procesos de Poisson

Patricia Kisbye

FaMAF

27 de abril, 2010

Método polar

- Con este método se generan dos variables normales independientes.

Si X e Y son normales estándar independientes, entonces

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

determinan variables R^2 y Θ independientes.

$$\begin{aligned} f(x,y) &= \frac{1}{2\pi}\exp(-(x^2+y^2)/2) \\ f(d,\theta) &= \frac{1}{2}\frac{1}{2\pi}e^{-d/2}, \quad 0 < d < \infty,\ 0 < \theta < 2\pi. \end{aligned}$$

R^2 y Θ resultan independientes, con $R^2 \sim \mathcal{E}(1/2)$ y $\Theta \sim \mathcal{U}(0, 2\pi)$.

Método polar

Método polar
Generar U_1, $U_2 \sim \mathcal{U}(0,1)$;
$R^2 \leftarrow -2\log(U)$;
$\Theta \leftarrow 2\pi U_2$;
$X \leftarrow R\cos(\Theta)$;
$Y \leftarrow R\,\mathrm{sen}(\Theta)$

Transformaciones de Box-Muller:

- $X \leftarrow \sqrt{-2\log(U_1)}\cos(2\pi U_2)$
- $Y \leftarrow \sqrt{-2\log(U_1)}\,\mathrm{sen}(2\pi U_2)$

- El cálculo de funciones trigonométricas lo hace poco eficiente.
- Se puede mejorar generando uniformes V_1 y V_2 en el círculo unitario.

Método polar

- Generar un par (V_1, V_2) uniformemente distribuido en el círculo unitario.
- $V_1^2 + V_2^2$ resulta uniforme en $(0, 1)$.
- $\Theta = \arctan V_2/V_1$ es uniforme en $(0, 2\pi)$.

Transformaciones de Box Muller

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

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

- $S = V_1^2 + V_2^2$ y Θ son independientes, luego podemos tomar

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

Método polar

Transformaciones de Box Muller

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

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

Método polar (última versión)

---

repeat

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

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

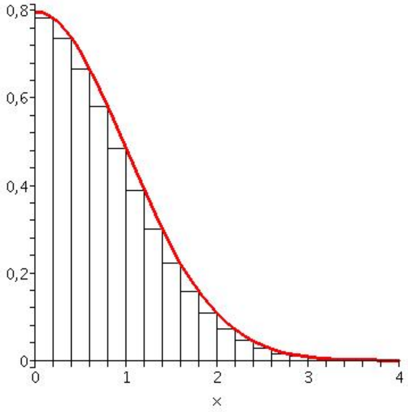
until $S < 1$;

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

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

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 $X_1 + X_2 + \cdots + X_j$, 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 ← 0, I ← 0;
while true do
    Generar U ~ U(0,1);
    if t − (1/λ) log U > T then
        stop
    else
        t ← t − (1/λ) log U;
        I ← I + 1;
        S[I] ← t
    end
end
```

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

Proceso de Poisson homogéneo

Generación de los primeros tiempos de arribo hasta $t = T$:
$N(T)$: 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 $U_1, U_2, \ldots, U_n$.
- Los tiempos de arribo son: $TU_1, TU_2, \ldots, TU_n$.
- Notar que deben ordenarse 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,
- 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.

Proceso de Poisson no homogéneo

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

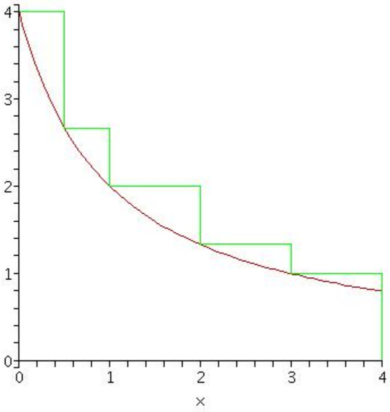
$t \leftarrow 0$, $I \leftarrow 0$;
while ***true*** **do**
    Generar $U \sim \mathcal{U}(0,1)$;
    if $t - \frac{1}{\lambda}\log U > T$ **then**
        stop
    else
        $t \leftarrow t - \frac{1}{\lambda}\log U$;
        Generar V;
        if $V < \lambda(t)/\lambda$ **then**
            $I \leftarrow I + 1$;
            $S[I] \leftarrow t$
        end
    end
end

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

Proceso de Poisson no homogéneo

Método de adelgazamiento o muestreo aleatorio

Un mejora consiste en particionar el intervalo $(0, T)$ en subintervalos, y aplicar el algoritmo anterior en cada uno de ellos:
$0 = t_0 < t_1 < \cdots < t_k < t_{k+1} = T$



$\lambda(s) \leq \lambda_j$,
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
        $X \leftarrow (X - t_J + t)\frac{\lambda_J}{\lambda_{J+1}}$;
        $t \leftarrow t_J$;
        $J \leftarrow J + 1$
    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):

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

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: Disminuye el número promedio de uniformes.

Proceso de Poisson no homogéneo

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

- $S_1, S_2, \ldots$: 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

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

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

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

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

Proceso de Poisson no homogéneo

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

Los sucesivos tiempos de eventos son entonces:

$$\begin{aligned}
S_1 &= \frac{3U_1}{1-U_1}\\
S_2 &= S_1 + (S_1+3)\frac{U_2}{1-U_2} = \frac{S_1+3U_2}{1-U_2}\\
&\vdots\\
S_j &= S_{j-1} + (S_{j-1}+3)\frac{U_{j-1}}{1-U_{j-1}} = \frac{S_{j-1}+3U_j}{1-U_j}
\end{aligned}$$

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
