Generando dígitos aleatorios e integración Monte Carlo.

A continuación se describe un procedimiento para generar dígitos pseudo-aleatorios desde una distribución objetivo mediante el método conocido como de transformación inversa.

Supondremos que es posible generar dígitos aleatorios desde la distribución $U(0,1)$. (este supuesto bastante habitual y está disponible en todos los computadores en la actualidad)

El método se basa en el siguiente resultado:

RESULTADO Sea $X \sim f_X(x)$ y defina la variable aleatoria $U = F_X(X)$ entonces.

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

(<u>OBS</u>: Si F_X es estrictamente creciente. Entonces $F_X^{-1}(u) = x \iff F_X(x) = u$. Si F_X es constante en algún intervalo, entonces

$$F_X^{-1}(u) = \inf \{x : F_X(x) \geq u\}$$)

<u>dem</u>: (caso estrictamente creciente) Note que para $U = F_X(X)$ sigue que $(0 < u < 1)$

$$P(U \leq u) = P(F_X(X) \leq u)$$

$$= P(F_X^{-1}(F_X(X)) \leq F_X^{-1}(u))$$

$$= P(X \leq F_X^{-1}(u)) = F_X(F^{-1}(u))$$

$$= u$$

Es decir

$$P(U \leq u) = u \quad , \quad 0 < u < 1.$$

(En efecto,

$$f_U(u) = F_U'(u) = \frac{d}{du} u = 1)$$

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

El resultado anterior permite generar dígitos aleatorios desde X. a través del siguiente procedimiento.

<u>Algoritmo:</u> Método de transformación inversa

1. Generar un número aleatorio $u \sim U(0,1)$.
2. Hacer
$$x = F_X^{-1}(u).$$
3. Retornar. $x \sim f_X$.

<u>Ejemplo:</u> (Generación de v.a. Exponenciales)

Considere X v.a. con función de densidad.

$$f(x) = e^{-x}, \quad x > 0. \qquad (*)$$

Es fácil notar que $F(x) = 1 - e^{-x}$. De este modo, podemos resolver

$$u = 1 - e^{-x},$$

para x, es decir

$$x = -\log(1-u).$$

Note además que $U \overset{d}{=} 1-U$ (en efecto, ambas U y $1-U$ son $U(0,1)$). De este modo el algoritmo puede ser escrito como sigue:

1. Generar un número aleatorio $u \sim U(0,1)$.
2. Hacer

$$x = -\log(u).$$

3. Retornar x como una v.a generada desde (*)

Una aplicación interesante de los métodos para simular dígitos aleatorios corresponde a la aproximación de integrales, método que es conocido como integración Monte Carlo. El objetivo es aproximar integrales del tipo

$$E_f\{h(x)\} = \int_{\mathcal{X}} h(x) f(x)\, dx$$

donde $\mathcal{X}$ es el espacio muestral

El método es basado en construir una muestra aleatoria $\{x_1, x_2, .., x_n\}$ generada desde la densidad f. Esto permite aproximar $E\{h(x)\}$ mediante el promedio empírico.

$$\overline{h}_n = \frac{1}{n} \sum_{k=1}^{n} h(x_k).$$

El método es exitoso debido a la Ley de los grandes números la que establece que

$$\overline{h}_n \xrightarrow{a.s} E_f\{h(x)\}$$

conforme $n \to \infty$. (discutiremos más adelante este tipo de nociones de convergencia de variables aleatorias). En efecto, podemos decir que $\overline{h}_n$ converge a $E_f\{h(x)\}$ con probabilidad 1

OBS: El problema de aproximación de integrales usando el método de Monte Carlo, puede ser expresado como un problema de estimación con parámetro de interés.

$$\theta = \int_{\chi} h(x) f(x)\,dx$$

- De ahí que este procedimiento "estima" θ, basado en datos "simulados" $x_1, \dots, x_m$, como

$$\hat{\theta} = \overline{h}_m = \frac{1}{m}\sum_{k=1}^{m} h(x_k)$$

y utilizar toda las herramientas de inferencia estadística.

- Adicionalmente, aunque este procedimiento es aproximado, podemos mejorar el error de aproximación simplemente simulando un número mayor de dígitos aleatorios. Una característica interesante del método es que puede ser generalizado con facilidad para aproximar integrales múltiples.

Ejemplo. Considere que deseamos calcular

$$\int_0^1 e^{-x}\,dx$$

Evidentemente podemos escribir esta integral como

$$E_f(e^{-x}) = \int_0^1 e^{-x}\,dx$$

con f. la función de densidad de una v.a $U(0,1)$. Así, suponga $x_1,..,x_n$ dígitos aleatorios simulados desde $U(0,1)$. Luego

$$\bar{h}_n = \frac{1}{n}\sum_{k=1}^{n} \exp(-x_k)$$

Ahora, note que

$$\int_0^1 e^{-x}\,dx = -e^{-x}\Big|_0^1 = -e^{-1}-(-e^{-0})$$

$$= 1-e^{-1} = 0.6321206$$

Considere el siguiente fragmento de codigo en R.

```
n <- 10 000
x <- runif(n)
z <- mean(exp(-x))
z
```

- [1] 0.6356812

Con $n = 10\,0000$ obtenemos $\bar{h}_n = 0.6324806$.

Suponga que deseamos calcular

$$\int_a^b h(x)\,dx$$

- una alternativa es considerar un cambio de variables de tal manera que los límites de integración sean desde 0 a 1. Ello es.

$$y = \frac{x-a}{b-a} \quad , \quad dy = \frac{1}{b-a} \cdot dx$$

Substituyendo

$$\int_a^b h(x)\,dx = \int_0^1 h(y(b-a)+a)\,(b-a)\,dy$$

Aún otra alternativa es reemplazar la densidad $U(0,1)$ por cualquier otra densidad definida en un intervalo dado por los límites de integración. Esto es.

$$\int_a^b h(x)\,dx = (b-a)\int_a^b h(x)\cdot\frac{1}{b-a}\,dx$$

es $(b-a)$ veces el valor esperado de $h(x)$ donde $x \sim U(a,b)$. Considere el siguiente ejemplo

Ejemplo. Suponga que deseamos calcular

$$\int_2^4 e^{-x}\,dx$$

Es decir, podemos hacer

$$\int_2^4 e^{-x}\,dx = (4-2)\int_2^4 e^{-x}\,\frac{1}{4-2}\,dx$$

Considere $x_1, \ldots, x_n$ dígitos aleatorios desde $U(2,4)$ y aproximamos.

$$\bar{h}_n = \frac{1}{n}\sum_{k=1}^{n} \exp(-x_k).$$

finalmente

$$\int_2^4 e^{-x}\,dx \approx (4-2)\cdot\frac{1}{n}\sum_{k=1}^{n} \exp(-x_k).$$

$$= 2\cdot\bar{h}_n.$$

El siguiente código en R permite aproximar la integral anterior

```
n <- 10000
x <- runif(n, min=2, max=4)
2 <- 2 * mean(exp(-x))
```

El valor exacto es

$$\int_2^4 e^{-x}\,dx = -e^{-x}\Big|_2^4 = e^{-2} - e^{-4}$$

$$= 0.1170196$$

por otro lado $2\cdot \overline{h}_n = 0.1175434$

Podemos resumir el procedimiento anterior para aproximar la integral

$$I = \int_a^b h(x)\,dx$$

Monte Carlo simple

1. Generar $x_1, \ldots, x_n$ v.a. IID desde $U(a,b)$

2. Calcular
$$\overline{h}_n = \frac{1}{n}\sum_{k=1}^{n} h(x_k)$$

3. Retornar
$$\hat{I} = (b-a)\,\overline{h}_n.$$

Suponga que deseamos calcular, la integral

$$\int_A h(x)\,dx$$

entonces, podemos escribir lo anterior como

- $$\int_A \frac{h(x)}{f(x)} f(x)\,dx = E_f\left\{\frac{h(x)}{f(x)}\right\}$$

donde la densidad $f(x)$ está definida en A (esto es $f(x) \geq 0 \ \forall x \in A$ y $\int_A f(x)\,dx = 1$). Entonces, podemos aproximar

- $$\int_A h(x)\,dx \approx \frac{1}{n}\sum_{k=1}^{n} \frac{h(x_k)}{f(x_k)}$$

con $x_1, \ldots, x_n$ v.a. IID desde la densidad f

Ejemplo. Supongamos que deseamos calcular

$$E\{\operatorname{sen}^2(x)\}$$

donde $x \sim N(0,1)$. Obtener un valor exacto analíticamente puede ser difícil mientras que una aproximación Monte Carlo es simple.

Es decir

$$E_{\phi}\{\operatorname{sen}^2(x)\} \approx \frac{1}{n}\sum_{k=1}^{n}\operatorname{sen}^2(x_k)$$

donde $x_1, \ldots, x_n$ son una muestra aleatoria generada desde $N(0,1)$. Por ejemplo podemos hacer

```
n <- 10 000
x <- rnorm(n)
z <- mean(sen(x)^2)
z
```

Escaneado con CamScanner

Monte Carlo puede ser usado para estimar probabilidades. En efecto para X una variable aleatoria tenemos

$$P(X \in A) = E\{I_A(X)\}$$

$$\approx \frac{1}{n} \sum_{k=1}^{n} I_A(X_k)$$

para $X_1, \ldots, X_n$ simulados desde la distribución de X para n suficientemente grande.

Ejemplo: Sea $X \sim N(0,1)$ y $a \in \mathbb{R}$. Entonces $p = P(X \leq a)$ no puede ser calculado explícitamente, pero

$$p_n = \frac{1}{n} \sum_{k=1}^{n} \mathbb{I}_{(-\infty, a]}(X_k)$$

puede ser usado como una aproximación para p.

Note que

$$I_{(-\infty,a]}(x_k) = \begin{cases} 1, & x_k \in (-\infty, a] \\ 0, & x_k \notin (-\infty, a] \end{cases}$$

$$= \begin{cases} 1, & x_k \leq a \\ 0, & x_k > a \end{cases}$$

es decir, podemos escribir

$$f_n = \frac{1}{n} \sum_{k=1}^{n} I_{\{x_k \leq a\}}.$$

Sea $a = 1.96$ y considere el siguiente código.

```
n <- 10000
x <- rnorm(n)
p <- mean(x < 1.96)
p
[1] 0.9762
```

mientras que el valor exacto es. 0.9750021