

EST-24107: Simulación

Profesor: Alfredo Garbuno Iñigo — Primavera, 2022 — Reducción de varianza.

Objetivo: Que veremos.

Lectura recomendada: Referencia.

1. INTRODUCCIÓN

Ya vimos algunos casos donde podemos reducir la varianza de nuestros estimadores Monte Carlo. Esto nos ayuda a mejorar la velocidad y eficiencia estadística de nuestros estimadores y en consecuencia optimizar los recursos computacionales.

Para lograrlo, por ejemplo, consideramos la posibilidad de cambiar la densidad contra la que estamos realizando el proceso de integración. A esta distribución le llamaremos **medida de referencia**.

En esta sección estudiaremos técnicas que nos permitirán reducir la varianza de nuestros estimadores.

1.1. Error Monte Carlo

Lo que estamos tratando de resolver es el problema de

$$\theta = \mathbb{E}_{\pi}(h(X)), \quad (1)$$

por medio de

1. Generar muestras $X_1, \dots, X_N \stackrel{\text{iid}}{\sim} \pi$.
2. Estimar por medio de $\hat{\theta}_N = (1/N) \sum_{i=1}^N h(X_i)$.

Bajo ciertas condiciones, un intervalo de confianza $(1 - \delta)$ puede construirse por medio

$$[\hat{\theta}_N - z_{1-\delta/2} \text{ee}(\hat{\theta}_N), \hat{\theta}_N + z_{1-\delta/2} \text{ee}(\hat{\theta}_N)], \quad (2)$$

donde podemos calcular el error estándar del estimador.

Hemos jugado con la noción de medir la calidad de nuestro estimador Monte Carlo al observar la longitud del intervalo. Es por esto que utilizamos la **longitud media (HW)** del intervalo de confianza

$$\text{HW} = z_{1-\delta/2} \text{ee}(\hat{\theta}_N). \quad (3)$$

Veremos técnicas de reducción de varianza que nos ayudarán a reducir la longitud media.

El uso de técnicas de reducción de varianza nos obligan a conocer un poco más sobre el modelo que está detrás de nuestro estimador. Esto es, para mejorar nuestra estimación $\hat{\pi}_N^{\text{MC}}(f)$ tenemos que conocer más propiedades tanto de f y/o π . Pues esto nos ayudará a reducir aún mas nuestra varianza.

2. VARIABLES ANTITÉTICAS

- Lo que buscaremos es inducir una correlación negativa entre secuencias de números pseudo-aleatorios.
- La idea es que al generar números en pares una observación grande en la primera secuencia se compense con una observación pequeña en la segunda.
- El ejemplo típico es sincronizar $u_n \sim \text{U}(0, 1)$ con $u'_n = 1 - u_n$.

2.1. Fundamento

Supongamos que tenemos $(X_1^{(1)}, \dots, X_N^{(1)})$ y $(X_1^{(2)}, \dots, X_N^{(2)})$ en donde para generar $X_j^{(1)}$ se utilizó u_j y para generar $X_j^{(2)}$ se utilizó $1 - u_j$.

2.1.1. Preguntas:

1. ¿Cuál es el valor esperado de $X_j^{(1)}$ y $X_j^{(2)}$?
2. ¿Son independientes?
3. ¿Qué pasa con los pares $(X_j^{(1)}, X_j^{(2)})$ y $(X_k^{(1)}, X_k^{(2)})$?

2.1.2. Conclusiones: Por lo anterior, si definimos

$$X_j = \frac{X_j^{(1)} + X_j^{(2)}}{2}, \quad \bar{X}_N = \frac{1}{N} \sum_{n=1}^N X_n, \quad (4)$$

tenemos un estimador que tiene la siguientes propiedades:

1. Es insesgado.
2. Tiene menor varianza que una muestra de $2N$ simulaciones.

2.1.3. *Consideraciones:* No siempre se puede lograr el objetivo. Es decir, depende del modelo.

2.2. Ejemplo: Integral en intervalo

Queremos estimar $\int_a^b f(x)dx$. El estimador Monte Carlo sería

$$\hat{\pi}_N^{\text{MC}}(f) = \frac{b-a}{N} \sum_{n=1}^N f(x_n), \quad (5)$$

donde $x_n \sim U(a, b)$.

Si escogemos la mitad (aleatoriamente) y por cada muestra usamos $x'_n = a + (b - x_n)$. Entonces tendríamos

$$\hat{\pi}_N^{\text{AMC}}(f) = \frac{b-a}{N/2} \sum_{n=1}^{N/2} \frac{f(x_n) + f(x'_n)}{2}, \quad (6)$$

```
1 set.seed(108)
2 nsamples <- 10^3;
3 a <- 2; b <- 3;
4 u <- runif(nsamples, min = a, max = b)
5 x <- dnorm(u)
```

```
1 estimador error.std      N
2 2.153e-02 4.413e-04 1.000e+03
```

```
1 u_ <- a + (b - u)
2 x_ <- dnorm(u_)
3 x <- (x + x_)/2
4 ax <- x[1:(nsamples/2)]
```

```
1 estimador error.std      N
2 2.133e-02 1.113e-04 5.000e+02
```

3. VARIABLES DE CONTROL

Supongamos que queremos estimar $\mathbb{E}(X)$ y tenemos acceso a una variable aleatoria Y que está **correlacionada** y se conoce $\nu = \mathbb{E}(Y)$. A Y se le conoce como **variable control** de X .

Sea $X_c = X - a(Y - \nu)$. Entonces

1. $\mathbb{E}(X_c) = \mathbb{E}(X)$.
2. $\mathbb{V}(X_c) = \mathbb{V}(X - a(Y - \nu)) = \mathbb{V}(X) + a^2\mathbb{V}(Y) - 2a\text{Cov}(X, Y)$. Esto implica que

$$\mathbb{V}(X_c) \leq \mathbb{V}(X) \quad \text{si} \quad 2a\text{Cov}(X, Y) > a^2\mathbb{V}(Y). \quad (7)$$

3. El caso particular

$$a^* = \frac{\text{Cov}(X, Y)}{\mathbb{V}(Y)}, \quad (8)$$

que induce la mínima varianza.

4. En este último caso

$$\mathbb{V}(X_c) = (1 - \rho_{X,Y}^2)\mathbb{V}(X). \quad (9)$$

3.1. Consideraciones:

En la práctica no siempre se conoce el valor de $\mathbb{V}(Y)$ y muy difícilmente la $\text{Cov}(X, Y)$, lo que implica que es difícil conocer el valor de a .

En la práctica se puede utilizar un estudio piloto para estimar a [1]. Esto es,

$$\hat{a}_M = \frac{\widehat{\text{Cov}}_M(X, Y)}{\widehat{\mathbb{V}}_M(Y)}. \quad (10)$$

Nota que el estimador resultante para la media de X_c ya no es un estimador insesgado.

3.2. Ejemplo

Supongamos que $X \sim \mathcal{N}(0, 1)$ y que $f(X) = \frac{X^6}{1+X^2}$.

- Entonces, utilizando la igualdad

$$\frac{x^6}{1+x^2} = x^4 - x^2 + 1 - \frac{1}{1+x^2}, \quad (11)$$

y podemos aproximar con $Y = g(X) = x^4 - x^2 + 1$.

- Para esta elección tenemos $\mathbb{E}(Y) = 3$.
- Así que el problema se reduce a

$$\mathbb{E}\left[\frac{X^6}{1+X^2}\right] = 3 - \mathbb{E}\left[\frac{1}{1+X^2}\right]. \quad (12)$$

```
1 set.seed(108)
2 x <- rnorm(nsamples)
```

```
1 f_x <- x**6/(1 + x**2)
2 c(estimador = mean(f_x), error.std = sd(f_x)/sqrt(nsamples))
```

```
1 estimador error.std
2 2.2788 0.2135
```

```
1 g_x ← 3 - 1 / (1 + x**2)
2 c(estimador = mean(g_x), error.std = sd(g_x)/sqrt(nsamples) )
```

```
1 estimador error.std
2 2.343346 0.008549
```

3.2.1. *Pregunta:* ¿Por qué estos estimadores dan los mismos números que con el código anterior?

```
1 set.seed(108)
2 x ← rnorm(100 * nsamples)
3 x ← array(x, c(100, nsamples))
4 f_x ← x**6/(1 + x**2)
5 estimadores ← apply(f_x, 1, mean)
6 c(estimador = mean(estimadores), error.std = sd(estimadores))
```

```
1 estimador error.std
2 2.3473 0.2752
```

```
1 g_x ← 3 - 1/(1+x**2)
2 estimadores ← apply(g_x, 1, mean)
3 c(estimador = mean(estimadores), error.std = sd(estimadores))
```

```
1 estimador error.std
2 2.3453 0.0081
```

4. MONTE CARLO CONDICIONAL

Se pueden utilizar algunos resultados teóricos intermedios para algunos casos. A esta técnica también se le conoce como método Rao-Blackwell.

Supongamos que nos interesa $\mathbb{E}(f(X))$ y del alguna manera tenemos conocimiento de una variable aleatoria que está relacionada con la original por medio de $\mathbb{E}(f(X)|Z = z)$. Utilizando la propiedad torre podemos calcular

$$\mathbb{E}(f(X)) = \mathbb{E}(\mathbb{E}(f(X)|Z = z)) . \quad (13)$$

Donde además tenemos que

$$\mathbb{V}(f(X)) = \mathbb{V}(\mathbb{E}(f(X)|Z)) + \mathbb{E}(\mathbb{V}(f(X)|Z)) . \quad (14)$$

Lo que buscamos es que:

1. Z pueda ser generado de manera eficiente.
2. Se pueda calcular $\mathbb{E}(f(X)|Z)$.

3. El valor de $\mathbb{E}(\mathbb{V}(f(X)|Z))$ sea grande.

Por lo tanto, el método es:

1. Generar una muestra $Z_1, \dots, Z_N \stackrel{\text{iid}}{\sim} \pi(Z)$.
2. Calcular $\mathbb{E}(f(X)|Z = z_k)$ de manera analítica.
3. Calcular el estimador de $\pi(f) = \mathbb{E}(f(X))$ por medio de

$$\hat{\pi}_N^{\text{CMC}}(f) = \frac{1}{N} \sum_{n=1}^N \mathbb{E}(f(X)|Z = Z_k). \quad (15)$$

4.1. Ejemplo: Mezcla Beta-Binomial

Supongamos un modelo Beta-Binomial. Igual que antes asumamos $n = 20$ y $\alpha = 2, \beta = 5$.

```
1 set.seed(108)
2 theta <- rbeta(nsamples, 2, 5)
3 y <- rbinom(nsamples, size = 20, theta)
4 c(estimador = mean(y), error.std = sd(y)/sqrt(nsamples))
```

```
1 estimador error.std
2 5.585 0.119
```

```
1 m_y <- 20 * theta
2 c(estimador = mean(m_y), error.std = sd(m_y)/sqrt(nsamples))
```

```
1 estimador error.std
2 5.587 0.102
```

El porcentaje de reducción de varianza es

```
1 [1] 0.1429
```

4.2. Ejemplo: Mezcla Poisson-Beta

Supongamos un modelo de mezcla

```
1 set.seed(108)
2 w <- rpois(nsamples, 10)
3 y <- rbeta(nsamples, w, w**2 + 1)
4 c(estimador = mean(y), error.std = sd(y)/sqrt(nsamples))
```

```
1 estimador error.std
2 0.096535 0.001404
```

```
1 m_y <- w / (w**2 + w + 1)
2 c(estimador = mean(m_y), error.std = sd(m_y)/sqrt(nsamples))
```

```

1 estimador error.std
2 0.098341 0.001019

```

El porcentaje de reducción de varianza es

```

1 [1] 0.2737

```

4.3. Ejemplo: Estimación de densidades

Podemos utilizar el método Monte Carlo condicionado para estimar densidades. Por ejemplo, si consideramos que $X_1, \dots, X_k \stackrel{\text{iid}}{\sim} \pi$ y nos interesa $S_k = X_1 + \dots + X_k$. Nos podemos preguntar por la densidad de la suma. Sabemos que la densidad es un objeto infinitesimal $\mathbb{P}(S_k \in dx)$. Y en algunas situaciones no tenemos acceso a éste.

Por ejemplo, consideremos $X_i \sim \text{Pareto}(1, \alpha = 3/2)$. Para este caso, no se puede escribir la densidad de S_k para $k > 1$. Lo que sí sabemos es que

$$S_k | S_{k-1} \stackrel{d}{=} X_k | S_{k-1} \sim \text{Pareto}(S_{k-1}, \alpha). \quad (16)$$

Por lo que podemos estimar la densidad de $X_k | S_{k-1}$ para valores, por ejemplo, en $[0, 15]$.

```

1 nsamples <- 5 * 10^3; ngrid <- 1000
2 rpareto <- function(n, alpha) {1 / runif(n)^(1/alpha) - 1}
3 dpareto <- function(x, alpha) { ifelse( x >= 0, (alpha / ((x+1)**(alpha + 1))),
4   0) }
5 k <- 4
6 u <- rpareto( (k-1) * nsamples, alpha = 3/2)
7 u <- array(u, c(k-1, nsamples))
8 S <- apply(u, 2, sum)
9 x <- seq(0.1, 15, length.out = ngrid)

```

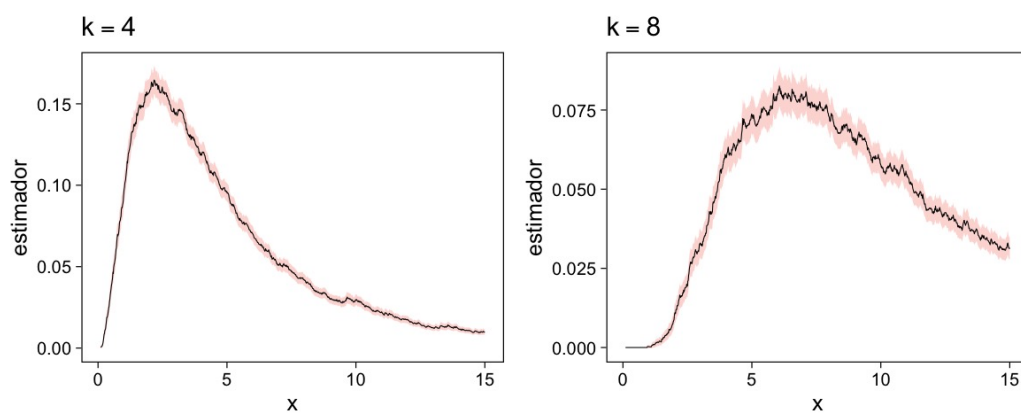
```

1 estimador <- array(x, c(ngrid,1)) >
2   apply(1, FUN = function(x_){ dpareto(x_ - S, alpha = 3/2) }) >
3   apply(2, mean)
4
5 error.std <- array(x, c(ngrid,1)) >
6   apply(1, FUN = function(x_){ dpareto(x_ - S, alpha = 3/2) }) >
7   apply(2, sd)

```

REFERENCIAS

- [1] S. S. Lavenberg, T. L. Moeller, and P. D. Welch. Statistical Results on Control Variables with Application to Queueing Network Simulation. *Operations Research*, 30(1):182–202, 1982. ISSN 0030-364X. [3](#)

FIGURA 1. Densidad de $x | S_{k-1}$.