

# Eficiencia de un estimador

**Definición 1** La eficiencia de un estimador  $\hat{\theta}(X)$  de un parámetro  $\theta$  de una variable aleatoria  $X$  es

$$\text{Eff}(\hat{\theta}(X)) \stackrel{\text{def}}{=} \frac{1}{\text{MSE}(\hat{\theta}(X)) \text{Coste}(\hat{\theta}(X))}$$

donde

$$\begin{aligned} \text{MSE}(\hat{\theta}(X)) &\stackrel{\text{def}}{=} \mathbb{E}[(\hat{\theta}(X) - \theta)^2] = \text{Var}(\hat{\theta}(X)) + \text{Sesgo}(\hat{\theta}(X), \theta)^2 \\ \text{Coste}(\hat{\theta}(X)) &\stackrel{\text{def}}{=} \text{tiempo esperado de cálculo de } \hat{\theta}(X) \end{aligned}$$

Dados dos estimadores insesgados,  $\hat{\theta}_1(X)$  y  $\hat{\theta}_2(X)$ , que requieren el mismo tiempo de cálculo, se tiene que

$$\text{Eff}(\hat{\theta}_1(X)) > \text{Eff}(\hat{\theta}_2(X)) \iff \text{Var}(\hat{\theta}_1(X)) < \text{Var}(\hat{\theta}_2(X))$$

Para  $\hat{\mu}_n(\mathbf{X})$  el estimador de Montecarlo de  $\mu = \mathbb{E}[g(\mathbf{X})]$  se tiene que

$$\text{Var}(\hat{\mu}_n(\mathbf{X})) = \frac{\sigma^2}{n} \quad \text{Sesgo}(\hat{\mu}_n(\mathbf{X}), \mu) = 0 \quad \text{Coste}(\hat{\mu}_n(\mathbf{X})) = cn$$

Por tanto,  $\text{Eff}(\hat{\mu}_n(\mathbf{X})) = \frac{1}{cn\sigma^2}$ . Es decir, al aumentar el tamaño de la muestra, la reducción en varianza se compensa exactamente con el incremento del tiempo de computación, por lo que no hay ganancia en eficiencia.

Para encontrar un estimador más eficiente se necesita, por tanto, reducir la varianza en un factor superior al incremento en tiempo de computación.

No hay una regla fija sobre cuán grande debe ser la mejora de la eficiencia proporcionada por un método de reducción de varianza para que valga la pena usarlo. En algunos escenarios, como la representación de gráficos por ordenador para películas animadas, donde miles de CPUs se mantienen ocupadas durante meses, una mejora del 10% aporta un ahorro significativo. En otros escenarios, como un cálculo único, una ganancia de factor 60 que convierte un minuto en un segundo de espera puede que no justifique el coste de la programación de un método más complicado.

La ganancia de eficiencia necesaria para justificar el uso de un método es menor si el esfuerzo de programación puede amortizarse en muchas aplicaciones. El umbral es alto para un programa de un solo uso, más bajo por algo que añadimos a nuestra biblioteca personal, más bajo aún para código que se comparte y aún más bajo para código que se incluye en una biblioteca o herramienta de simulación de uso general.