# Capítulo 9

# Métodos de Monte Carlo

El presente capítulo es un resumen de las principales ideas recogidas en el capítulo 29 de [MacKay, 2003].

## 9.1. Los problemas a resolver

Los métodos de Monte Carlo son técnicas computacionales que hacen uso de números aleatóreos. El propósito de los métodos de Monte Carlo es resolver uno o ambos de los siguientes problemas.

**Problema 1:** generar muestras $\{\boldsymbol{x}^{(r)}\}_{r=1}^{R}$ a partir de una distribución de probabilidad dada, $P(\boldsymbol{x})$.

**Problema 2:** estimar esperanzas de funciones bajo esta distribución, por ejemplo

$$\Phi = \mathrm{E}[\phi(\boldsymbol{x})] \equiv \int P(\boldsymbol{x})\phi(\boldsymbol{x})d^N\boldsymbol{x}. \tag{9.1}$$

La distribución de probabilidades $P(\boldsymbol{x})$, que se denomina *densidad objetivo*, podría ser una distribución procedente de la física estadística o una distribución condicional de las que surgen en modelado de datos — por ejemplo, la probabilidad a posteriori de unos parámetros del modelo dados algunos datos observados. En general, asumiremos que $\boldsymbol{x}$ es un vector $N$-dimensional con componentes reales $x_n$, pero a veces consideraremos también espacios discretos.

Ejemplos simples de funciones $\phi(\boldsymbol{x})$ cuyas esperanzas nos podrían interesar incluyen los momentos de primer y segundo orden de cantidades que deseamos predecir, a partir de los cuales podemos calcular medias y varianzas; por ejemplo, si cierta cantidad $t$ depende de $\boldsymbol{x}$, podemos encontrar la media y la varianza de $t$ bajo $P(\boldsymbol{x})$ hallando las esperanzas de las funciones $\phi_1(\boldsymbol{x}) = t(\boldsymbol{x})$ y $\phi_2(\boldsymbol{x}) = (t(\boldsymbol{x}))^2$,

$$\Phi_1 \equiv \mathrm{E}[\phi_1(\boldsymbol{x})] \text{ y } \Phi_2 \equiv \mathrm{E}[\phi_2(\boldsymbol{x})], \tag{9.2}$$

por tanto, se obtiene

$$\overline{t} = \Phi_1 \text{ y } \mathrm{var}(t) = \Phi_2 - \Phi_1^2. \tag{9.3}$$

Se asume que $P(\boldsymbol{x})$ es lo suficientemente compleja para que estas esperanzas no puedan evaluarse por métodos exactos; de modo que estamos interesados en los métodos de Monte Carlo.

Nos concentraremos en el primer problema (muestreo), porque si somos capaces de resolverlo, entonces podemos solucionar el segundo problema usando las muestras aleatóreas obtenidas $\{\boldsymbol{x}^{(r)}\}_{r=1}^{R}$. Éstas generan el siguiente estimador de la esperanza

$$\hat{\Phi} \equiv \frac{1}{R}\sum_{r}\phi(\boldsymbol{x}^{(r)}). \tag{9.4}$$

Si los vectores $\{\boldsymbol{x}^{(r)}\}_{r=1}^{R}$ se generan a partir de $P(\boldsymbol{x})$, entonces la esperanza de $\hat{\Phi}$ es $\Phi$. Además, conforme el número de muestras $R$ aumenta, la varianza de $\hat{\Phi}$ decrece como $\frac{\sigma^2}{R}$, donde $\sigma^2$ es la varianza de $\phi$,

$$\sigma^2 = \int P(\boldsymbol{x})(\phi(\boldsymbol{x}) - \Phi)^2 d^N\boldsymbol{x}. \tag{9.5}$$

Ésta es una de las propiedades importantes de los métodos de Monte Carlo.

> La exactitud de la estimación de Monte Carlo (9.4) depende solamente de la varianza de $\phi$, no de la dimensión del espacio muestreado. Para ser precisos, la varianza de $\hat{\Phi}$ va como $\frac{\sigma^2}{R}$. Así que, sin reparar en la dimensión de $\boldsymbol{x}$, puede ser que tan pocas muestras $\{\boldsymbol{x}^{(r)}\}$ como una docena basten para estimar $\Phi$ satisfactoriamente.

Posteriormente descubriremos, sin embargo, que la alta dimensionalidad puede causar otros problemas a los métodos de Monte Carlo. Obtener muestras independientes de una distribución $P(\boldsymbol{x})$ no es a menudo sencillo.

### 9.1.1. ¿Por qué es difícil obtener muestras de $P(\boldsymbol{x})$?

Asumiremos que la densidad a partir de la cual deseamos obtener muestras, $P(\boldsymbol{x})$, puede ser evaluada, salvo una constante multiplicativa; es decir, podemos evaluar una función $P^*(\boldsymbol{x})$ tal que

$$P(\boldsymbol{x}) = \frac{P^*(\boldsymbol{x})}{Z}. \tag{9.6}$$

Si podemos evaluar $P^*(\boldsymbol{x})$, ¿por qué no podemos resolver fácilmente el problema 1? ¿Por qué, en general, es difícil obtener muestras de $P(\boldsymbol{x})$? Existen dos dificultades. La primera es que habitualmente no conocemos la constante de normalización

$$Z = \int P^*(\boldsymbol{x}) d^N\boldsymbol{x}. \tag{9.7}$$

La segunda es que, incluso si conociéramos $Z$, el problema de extraer muestras de $P(\boldsymbol{x})$ sigue siendo un desafío, especialmente en espacios de alta dimensión, puesto que no hay

una forma obvia de muestrear $P$ sin enumerar todos o la mayoría de los posibles estados. Las muestras correctas de $P$ por definición tenderán a proceder de lugares en el espacio-$\boldsymbol{x}$ donde $P(\boldsymbol{x})$ es grande; ¿cómo podemos identificar aquellos lugares donde $P(\boldsymbol{x})$ es grande, sin evaluar $P(\boldsymbol{x})$ en todos sitios? Solamente existen unas pocas densidades multidimensionales a partir de las cuales es fácil extraer muestras, por ejemplo, la distribución gaussiana.

Comencemos con un ejemplo simple unidimensional. Imaginemos que deseamos extraer muestras de la densidad $P(x) = \frac{P^*(x)}{Z}$ donde

$$P^*(x) = \exp[0{,}4(x - 0{,}4)^2 - 0{,}08x^4], \; x \in (-\infty, \infty). \tag{9.8}$$

Podemos dibujar esta función, pero esto no quiere decir que podamos extraer muestras de ella. Para empezar, desconocemos la constante de normalización $Z$. Para afrontar un problema más simple, podríamos discretizar la variable $x$ y pedir muestras de la distribución discreta de probabilidades sobre un conjunto finito de puntos uniformemente espaciados $\{x_i\}$. ¿Cómo podríamos resolver este problema? Si evaluamos $p_i^* = P^*(x_i)$ en cada punto $x_i$, podemos calcular

$$Z = \sum_i p_i^* \tag{9.9}$$

y

$$p_i = \frac{p_i^*}{Z} \tag{9.10}$$

y entonces podríamos muestrear la distribución de probabilidades $\{p_i\}$ usando distintos métodos basados en una fuente de bits aleatóreos. Pero, ¿cuál es el coste de este procedimiento, y cómo se escala con la dimensión del espacio, $N$? Vamos a concentrarnos en el coste inicial de evaluar $Z$ (ecuación (9.9)). Para calcular $Z$ tenemos que visitar todos los puntos del espacio. Supongamos que hay 50 puntos uniformemente espaciados en una dimensión. Si nuestro sistema tuviera $N$ dimensiones, por ejemplo $N = 1000$, entonces el número de puntos correspondiente sería $50^{1000}$, un inimaginable número de evaluaciones de $P^*$. Incluso si cada componente $x_n$ tomara sólo dos valores discretos, el número de evaluacions de $P^*$ sería $2^{1000}$, un número que sigue siendo horriblemente enorme. Si cada electrón del universo (existen aproximadamente unos $2^{266}$ electrones) fuese un computador a 1000 gigahertzios que pudiese evaluar $P^*$ para un trillón ($2^{40}$) de estados cada segundo, y utilizáramos esos $2^{266}$ computadores durante un tiempo igual a la edad del universo ($2^{58}$ segundos), únicamente visitarían $2^{364}$ estados. Tendríamos que esperar que transcurriesen más de $2^{636} \propto 10^{190}$ edades del universo antes de que los $2^{1000}$ estados hayan sido visitados.

Existen numerosos sistemas de $2^{1000}$ estados. Un ejemplo es una colección de 1000 espines, como un fragmento $30 \times 30$ de un modelo *Ising* cuya distribución de probabilidades es proporcional a

$$P^*(\boldsymbol{x}) = \exp\left[-\beta E(\boldsymbol{x})\right] \tag{9.11}$$

donde $x_n \in \{\pm 1\}$ y

$$E(\boldsymbol{x}) = -\left[\frac{1}{2}\sum_{m,n} J_{mn} x_m x_n + \sum_n H_n x_n\right]. \tag{9.12}$$

La función de energía $E(\boldsymbol{x})$ es rápidamente evaluada para cualquier $\boldsymbol{x}$. Pero si deseamos evaluar esta función en *todos* los estados $\boldsymbol{x}$, el tiempo de cálculo requerido sería de $2^{1000}$ evaluaciones de función.

### 9.1.2. Una útil analogía

Imaginemos la tarea de extraer muestras aleatóreas de agua de un lago y encontrar la concentración media de plancton. La profundidad del lago en $\boldsymbol{x} = (x, y)$ es $P^*(\boldsymbol{x})$, y establecemos que la concentración de plancton es una función de $\boldsymbol{x}$, $\phi(\boldsymbol{x})$. La concentración media buscada es una integral como (9.1), es decir

$$\Phi = \mathrm{E}[\phi(\boldsymbol{x})] \equiv \frac{1}{Z} \int P^*(\boldsymbol{x}) \phi(\boldsymbol{x}) d^N \boldsymbol{x}, \tag{9.13}$$

donde $Z = \int P^*(\boldsymbol{x}) dxdy$ es el volumen del lago. Disponemos de una barca, de un sistema de navegación por satélite y de una plomada. Usando el navegador, podemos llevar la barca a cualquier ubicación deseada $\boldsymbol{x}$ sobre el mapa; utilizando la plomada, podemos medir $P^*(\boldsymbol{x})$ en ese punto. También podemos medir la concentración de plancton allí.

El problema 1 es extraer al azar muestras de $1cm^3$ de agua del lago, de tal forma que cada muestra sea igualmente probable de proceder de cualquier punto dentro del lago. El problema 2 es encontrar la concentración media de plancton.

Éstos son problemas difíciles de resolver porque al principio no conocemos nada acerca de la profundidad del lago $P^*(\boldsymbol{x})$. Quizás gran parte del volumen del lago esté contenida en cañones estrechos y profundos, en cuyo caso, para muestrear correctamente del lago y estimar $\Phi$ correctamente nuestro método debe implícitamente descubrir los cañones y encontrar su volumen relativo al resto del lago. Son problemas difíciles, sin embargo, veremos que los inteligentes métodos de Monte Carlo pueden resolverlos.

### 9.1.3. Muestreo uniforme

Habiendo aceptado que no podemos visitar exhaustivamente cada ubicación $\boldsymbol{x}$ del espacio de estados, podríamos intentar resolver el segundo problema (estimar la esperanza de una funcion $\phi(\boldsymbol{x})$) extrayendo muestras aleatóreas $\{\boldsymbol{x}^{(r)}\}_{r=1}^{R}$ uniformemente del espacio de estados y evaluando $P^*(\boldsymbol{x})$ en esos puntos. Por tanto, podríamos introducir una constante de normalización $Z_R$, definida por

$$Z_R = \sum_{r=1}^{R} P^*(\boldsymbol{x}^{(r)}), \tag{9.14}$$

y estimar $\Phi = \int P(\boldsymbol{x}) \phi(\boldsymbol{x}) d^N \boldsymbol{x}$ como

$$\hat{\Phi} = \sum_{r=1}^{R} \phi(\boldsymbol{x}^{(r)}) \frac{P^*(\boldsymbol{x}^{(r)})}{Z_R}. \tag{9.15}$$

¿Algo va mal con esta estrategia? Depende de las funciones $\phi(\boldsymbol{x})$ y $P^*(\boldsymbol{x})$. Asumamos que $\phi(\boldsymbol{x})$ es una función que varía suavemente, y concentrémonos en la naturaleza de

$P^*(\boldsymbol{x})$. Una distribución $P^*(\boldsymbol{x})$ de alta dimensión se concentra a menudo en una pequeña región del espacio de estados conocida como *conjunto típico* $T$, cuyo volumen está dado por $|T| \approx 2^{H(\boldsymbol{X})}$, donde $H(\boldsymbol{X})$ es la entropía de la distribución de probabilidades $P(\boldsymbol{x})$. Si casi toda la masa de probabilidad está localizada en el conjunto típico y $\phi(\boldsymbol{x})$ es una función benigna, el valor de $\Phi = \int P(\boldsymbol{x})\phi(\boldsymbol{x})d^N\boldsymbol{x}$ estará principalmente determinado por los valores que $\phi(\boldsymbol{x})$ tome en el conjunto típico. En consecuencia, el muestreo uniforme sólo constituirá una oportunidad de dar una buena estimación de $\Phi$ si hacemos el número de muestras $R$ lo suficientemente grande para que sea probable golpear el conjunto típico al menos una vez o dos. Por tanto, ¿cuántas muestras se requieren? Tomemos de nuevo el caso del modelo *Ising*. El tamaño total del espacio de estados es de $2^N$ estados, y el conjunto típico tiene un tamaño de $2^H$. Así que cada muestra tiene una probabilidad de $\frac{2^H}{2^N}$ de caer en el conjunto típico. El número de muestras requerido para golpear el conjunto típico al menos una vez es del orden de

$$R_{min} \approx 2^{N-H} \tag{9.16}$$

Nos preguntamos ahora el valor de $H$. A altas temperaturas, la distribución de probabilidades de un modelo *Ising* tiende a una distribución uniforme y la entropía tiende a $H_{max} = N$ bits, lo que significa que $R_{min}$ es del orden de 1. Bajo estas condiciones, el muestreo uniforme puede ser una técnica satisfactoria para la estimación de $\Phi$. Pero las altas temperaturas no son de gran interés. Considerablemente más interesantes son las temperaturas intermedias, como la temperatura crítica a la cual el modelo *Ising* pasa de una fase ordenada a una fase desordenada. La temperatura crítica de un modelo *Ising* infinito es $\theta_c = 2{,}27$. A esta temperatura, la entropía de un modelo *Ising* es aproximadamente $\frac{N}{2}$ bits. Para esta distribución de probabilidades, el número de muestras requeridas para golpear el conjunto típico una vez es del orden de

$$R_{min} \approx 2^{N-\frac{N}{2}} = 2^{\frac{N}{2}}, \tag{9.17}$$

que, para $N = 1000$, es de $10^{150}$. Éste es aproximadamente el cuadrado del número de partículas del universo. Por tanto, el muestreo uniforme es totalmente inservible para el estudio de modelos *Ising* de tamaño modesto. Y en problemas de más alta dimensión, si la distribución $P(\boldsymbol{x})$ no es realmente uniforme, es improbable que el muestreo uniforme sea útil.

Habiendo establecido que extraer muestras de una distribución de alta dimensión $P(\boldsymbol{x}) = P^*(\boldsymbol{x})/Z$ es difícil incluso si $P^*(\boldsymbol{x})$ es fácil de evaluar, ahora estudiaremos una secuencia de métodos de Monte Carlo más sofisticados: *muestreo por importancia, muestreo por rechazo, método de Metropolis, muestreo de Gibbs, y muestreo por rodajas.*

## 9.2. Muestreo por importancia

El muestreo por importancia no es un método para generar muestras de $P(\boldsymbol{x})$ (problema 1); es un método para estimar la esperanza de una función $\phi(\boldsymbol{x})$ (problema 2). Puede verse como una generalización del método de muestreo uniforme.

Imaginemos que la distribución objetivo es una densidad unidimensional $P(x)$. Asumamos que somos capaces de evaluar esta densidad en cualquier punto $\boldsymbol{x}$, salvo una constante

multiplicativa; por tanto, podemos evaluar una función $P^*(x)$ tal que

$$P(x) = P^*(x)/Z. \tag{9.18}$$

Pero $P(x)$ es una función demasiado complicada para que seamos capaces de muestrearla directamente. Asumamos ahora que tenemos una densidad más simple $Q(x)$ a partir de la cual podemos generar muestras, y que también podemos evaluar salvo una constante multiplicativa (es decir, podemos evaluar $Q^*(x)$, donde $Q(x) = Q^*(x)/Z_Q$). Denominamos a $Q$ *densidad muestreadora*.

En el muestreo por importancia, generamos $R$ muestras $\{x^{(r)}\}$ de $Q(x)$. Si estos puntos fueran muestras de $P(x)$, entonces podríamos estimar $\Phi$ mediante la ecuación (9.4). Pero cuando generamos muestras de $Q$, los valores de $x$ donde $Q(x)$ es mayor que $P(x)$ estarán *sobrerrepresentados* en este estimador, y los puntos donde $Q(x)$ es menor que $P(x)$ estarán *subrepresentados.* Para tener en cuenta el hecho de que hemos muestreado de la distribución equivocada, introducimos los siguientes *pesos*

$$w_r \equiv \frac{P^*(x^{(r)})}{Q^*(x^{(r)})} \tag{9.19}$$

que se usan para ajustar la *importancia* de cada punto en nuestro estimador, por tanto:

$$\hat{\Phi} \equiv \frac{\sum_r w_r \phi(x^{(r)})}{\sum_r w_r}. \tag{9.20}$$

Una dificultad práctica del muestreo por importancia es la complejidad de calcular la fiabilidad del estimador $\hat{\Phi}$. La varianza del estimador es desconocida de antemano, porque depende de una integral respecto a $x$ de una función que involucra a $P^*(x)$. Y la varianza de $\hat{\Phi}$ es difícil de estimar, porque las varianzas empíricas de las cantidades $w_r$ y $w_r\phi(x^{(r)})$ no son una buena guía hacia las verdaderas varianzas del numerador y del denominador de la ecuación (9.20). Si la densidad propuesta $Q(x)$ es pequeña en una región donde $|\phi(x)P^*(x)|$ es grande, entonces es bastante posible, incluso despues de que se hayan generado muchos puntos $x^{(r)}$, que ninguno de ellos haya caído en esa región. En este caso, la estimación de $\Phi$ sería drásticamente errónea, y no habría ninguna indicación en la varianza *empírica* de que la varianza verdadera del estimador $\hat{\Phi}$ es grande.

De la práctica, se conoce que la densidad muestreadora debería tener **colas importantes**. Por ejemplo, se prefiere el empleo de una densidad muestreadora de Cauchy frente a una gaussiana. El estimador $\hat{\Phi}$ obtenido con la densidad muestreadora de Cauchy convergerá al valor verdadero $\Phi$ con menos muestras $R$ que si usamos una gaussiana.

### 9.2.1. Muestreo por importancia en espacios multidimensionales

Ya hemos observado que se necesita cuidado en los problemas unidimensionales de muestreo por importancia. Ahora nos preguntamos si el muestreo por importancia es una técnica útil para espacios de mayor dimensión, por ejemplo $N = 1000$.

Consideremos un caso simple donde la densidad objetivo $P(\boldsymbol{x})$ es una distribución uniforme dentro de una esfera,

$$P^*(\boldsymbol{x}) = \begin{cases} 1 & \text{si} \quad 0 \leq \rho(\boldsymbol{x}) \leq R_P \\ 0 & \text{si} \quad \rho(\boldsymbol{x}) > R_P, \end{cases} \tag{9.21}$$

donde $\rho(\boldsymbol{x}) \equiv (\sum_i x_i^2)^{1/2}$, y la densidad muestreadora es una gaussiana centrada en el origen,

$$Q(\boldsymbol{x}) = \prod_i \text{Normal}(x_i; 0, \sigma^2). \tag{9.22}$$

Un método de muestreo por importancia dará problemas si el estimador $\hat{\Phi}$ está dominado por unos pocos pesos $w_r$ de valores grandes. ¿Cuál será el rango típico de los valores de los pesos $w_r$? Sabemos que si $\rho$ es la distancia al origen de una muestra de $Q$, la cantidad $\rho^2$ tiene aproximadamente una distribución gaussiana con media y desviación típica:

$$\rho^2 \sim N\sigma^2 \pm \sqrt{2N}\sigma^2. \tag{9.23}$$

Por tanto, casi todas las muestras de $Q$ están en un conjunto típico que dista del origen aproximadente $\sqrt{N}\sigma$. Asumamos que $\sigma$ se elige de tal forma que el conjunto típico de $Q$ resida dentro de la esfera de radio $R_P$. (Si no se elige así, entonces las leyes de los grandes números implican que casi todas las muestras generadas de $Q$ caerán fuera de $R_P$ y tendrán peso nulo.) Por tanto, sabemos que la mayoría de las muestras de $Q$ tendrán un valor de $Q$ que está en el rango

$$\frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{N}{2} \pm \frac{\sqrt{2N}}{2}\right). \tag{9.24}$$

En consecuencia, los pesos $w_r = P^*/Q = 1/Q$ tendrán típicamente valores en el rango

$$(2\pi\sigma^2)^{N/2} \exp\left(\frac{N}{2} \pm \frac{\sqrt{2N}}{2}\right). \tag{9.25}$$

De modo que si extraemos un centenar de muestras, cuál serál el rango típico de los pesos. Podemos estimar aproximadamente la ratio entre el mayor peso y el peso mediano doblando la desviación estándar en la ecuación (9.25). El mayor peso y el mediano estarán típicamente en la ratio:

$$\frac{w_r^{max}}{w_r^{med}} = \exp\left(\sqrt{2N}\right). \tag{9.26}$$

En $N = 1000$ dimensiones, es probable que el mayor peso después de un centenar de muestras, sea aproximadamente $10^{19}$ veces mayor que el peso mediano. Por consiguiente, una estimación mediante muestreo por importancia para un problema de alta dimensión será muy probablemente totalmente dominada por unas pocas muestras con pesos enormes.

En resumen, el muestreo por importancia en altas dimensiones sufre a menudo dos dificultades. En primer lugar, necesitamos obtener muestras que residan en el conjunto típico de $P$, y esto puede llevar un largo tiempo, salvo que $Q$ sea una buena aproximación

de $P$. En segundo lugar, incluso si obtenemos muestras en el conjunto típico, es probable que los pesos asociados a esas muestras varíen en grandes factores, porque las probabilidades de los puntos en el conjunto típico difieren en factores del orden de $\exp(\sqrt{N})$, así que los pesos también diferirán, salvo que $Q$ sea una aproximación casi perfecta de $P$.

## 9.3. Muestreo por rechazo

Asumamos de nuevo una densidad unidimensional $P(x) = P^*(x)/Z$ que es una función demasiado compleja para que seamos capaces de muestrearla directamente. Asumamos que tenemos una *densidad propuesta* $Q(x)$ más simple, que podemos evaluar (salvo un factor multiplicativo $Z_Q$), y de la cual podemos generar muestras. Asumamos también que conocemos el valor de una constante $c$ tal que

$$cQ^*(x) > P^*(x), \text{ para todo } x. \tag{9.27}$$

Se generan dos números aleatóreos. El primero, $x$, se genera a partir de la densidad propuesta $Q(x)$. Evaluamos $cQ^*(x)$ y generamos una variable uniformemente distribuida $u$ en el intervalo $[0, cQ^*(x)]$.

Ahora evaluamos $P^*(x)$ y aceptamos o rechazamos la muestra $x$ comparando el valor de $u$ con el valor de $P^*(x)$. Si $u > P^*(x)$, entonces $x$ es rechazada; en caso contrario, es aceptada, lo que significa que añadimos $x$ a nuestro conjunto de muestras $\{x^{(r)}\}$. El valor de $u$ es desechado.

Nos podemos preguntar por qué este procedimiento genera muestras de $P(x)$. El punto propuesto $(x, u)$ procede con probabilidad uniforme del área bajo la curva $cQ^*(x)$. La regla de rechazo descarta todos los puntos que se ubiquen por encima de la curva $P^*(x)$. En consecuencia, los puntos $(x, u)$ que se aceptan están uniformemente distribuidos en el área bajo $P^*(x)$. Esto implica que la densidad de probabilidad de las coordenadas-$x$ de los puntos aceptados debe ser proporcional a $P^*(x)$, así que las muestras deben ser muestras independientes de $P(x)$.

El muestreo por rechazo trabaja mejor si $Q$ es una buena aproximación de $P$. Si $Q$ es muy diferente de $P$ entonces, para que $cQ$ exceda a $P$ en todos sitios, $c$ tendrá que ser necesariamente muy grande y la frecuencia de rechazo será también grande.

### 9.3.1. Muestreo por rechazo en espacios multidimensionales

En un problema de alta dimensión es muy probable que el requerimiento de que $cQ^*$ sea una cota superior de $P^*$ fuerce que $c$ sea tan enorme que la aceptación de los puntos será muy rara. Encontrar tal valor de $c$ puede también ser difícil, ya que en muchos problemas no sabemos ni dónde están localizados los máximos de $P^*$ ni cómo de grandes son.

Como caso de estudio, consideremos un par de distribuciones gaussianas N-dimensionales de media cero. Imaginemos que generamos muestras de una de ellas, cuya desviación típica es $\sigma_Q$, y que usamos muestreo por rechazo para obtener muestras de la otra, cuya desviación estándar es $\sigma_P$. Asumamos que estas dos desviaciones estándar están cercanas

en su valor — por ejemplo, $\sigma_Q$ es un 1 % mayor que $\sigma_P$. ($\sigma_Q$ debe ser mayor que $\sigma_P$, porque en caso contrario, no existirá $c$ tal que $cQ$ exceda a $P$ para todo $\boldsymbol{x}$.) Ahora la cuestión es qué valor de $c$ requeremos si la dimensión del espacio es $N = 1000$. La densidad de $Q(\boldsymbol{x})$ en el origen es $1/(2\pi\sigma_Q^2)^{N/2}$, así que para forzar que $cQ$ exceda a $P$ necesitamos establecer

$$c = \frac{(2\pi\sigma_Q^2)^{N/2}}{(2\pi\sigma_P^2)^{N/2}} = \exp\left(N \ln \frac{\sigma_Q}{\sigma_P}\right). \tag{9.28}$$

Con $N = 1000$ y $\frac{\sigma_Q}{\sigma_P} = 1{,}01$, se obtiene $c = \exp(10) \approx 20000$. Nos preguntamos ahora cuál será la tasa de aceptación para este valor de $c$. La respuesta es inmediata: como la tasa de aceptación es el ratio entre el volumen bajo la curva $P(\boldsymbol{x})$ y el volumen bajo $cQ(\boldsymbol{x})$, el hecho de que $P$ y $Q$ estén ambos normalizados implica que la tasa de aceptación será de $1/c$, por ejemplo, $1/20000$. En general, $c$ crece exponencialmente con la dimensión, $N$, así que la tasa de aceptación decrece exponencialmente con $N$.

Por tanto, el muestreo por rechazo, aunque es un método útil para problemas unidimensionales, no se espera que sea una técnica práctica para generar muestras de distribuciones $P(\boldsymbol{x})$ de alta dimensión.

## 9.4. El método de Metropolis-Hastings

El muestreo por importancia y el muestreo por rechazo trabajan bien sólo si la densidad propuesta $Q(x)$ es similar a $P(x)$. En problemas grandes y complejos es difícil crear una única densidad $Q(x)$ que tenga esta propiedad.

El algoritmo de Metropolis-Hastings en lugar de esto hace uso de una densidad propuesta $Q$ *la cual depende del estado actual* $x^{(t)}$. La densidad $Q(x'; x^{(t)})$ podría ser una distribución simple, tal como una gaussiana centrada en el actual $x^{(t)}$. La densidad propuesta $Q(x'; x)$ puede ser *cualquier* densidad fija a partir de la cual extraemos muestras. A diferencia del muestreo por importancia y del muestreo por rechazo, no es necesario que $Q(x'; x^{(t)})$ sea similar a $P(x)$ para que el algoritmo sea útil en la práctica.

Como antes, asumamos que podemos evaluar $P^*(x)$ para cualquier $x$. Un nuevo estado tentativo $x'$ se genera a partir de la densidad propuesta $Q(x'; x^{(t)})$. Para decidir si aceptamos el nuevo estado, calculamos la cantidad

$$a = \frac{P^*(x')}{P^*(x^{(t)})} \frac{Q(x^{(t)}; x')}{Q(x'; x^{(t)})}. \tag{9.29}$$

Si $a \geq 1$, entonces el nuevo estado es aceptado.

En otro caso, el nuevo estado es aceptado con probabilidad $a$.

Si el paso es aceptado, establecemos $x^{(t+1)} = x'$.

Si el paso es rechazado, entonces establecemos $x^{(t+1)} = x^{(t)}$.

Nótese la diferencia con el muestreo por rechazo: en el muestreo por rechazo, los puntos rechazados eran descartados y no tenían influencia en la lista de muestras $\{x^{(r)}\}$ que confeccionábamos. Aquí, un rechazo causa que el estado actual sea escrito otra vez en la lista de muestras.

**Notación.** Hemos usado el superíndice $r = 1, \ldots, R$ para etiquetar puntos que son muestras *independientes* de una distribución, y el superíndice $t = 1, \ldots, T$ para etiquetar la secuencia de estados de una cadena de Markov. Es importante notar que una simulación de Metropolis-Hastings de $T$ iteraciones no produce $T$ muestras *independientes* de la distribución objetivo $P$. Las muestras están correladas.

Para calcular la probabilidad de aceptación (ecuación (9.29)), necesitamos ser capaces de computar los ratios $P(x')/P(x^{(t)})$ y $Q(x^{(t)}; x')/Q(x'; x^{(t)})$. Si la densidad propuesta es una simple densidad simétrica, tal como una gaussiana centrada en el punto actual, entonces el segundo ratio es la unidad, y el método de Metropolis-Hastings simplemente involucra comparar los valores de la densidad objetivo en los dos puntos. Este caso especial es a veces llamado el método de Metropolis. Sin embargo, denominaremos al algoritmo general de Metropolis-Hastings para $Q$ asimétricas como 'método de Metropolis', ya que las ideas importantes merecen nombres cortos.

### 9.4.1. Convergencia del método de Metropolis hacia la densidad objetivo

Puede demostrarse que para cualquier $Q$ positiva (es decir, cualquier $Q$ tal que $Q(x'; x) > 0$, para todo $x, x'$), conforme $t \to \infty$, la distribución de probabilidad de $x^{(t)}$ tiende a $P(x) = P^*(x)/Z$. (Esta afirmación no implica que $Q$ tenga que asignar probabilidades positivas para cada punto $x'$; nótese también que no hemos dicho nada acerca de cómo de rápida tiene lugar la convergencia hacia $P(x)$.)

El método de Metropolis es un ejemplo de método de *Monte Carlo con Cadenas de Markov* (abreviado MCMC). A diferencia del muestreo por rechazo, donde los puntos aceptados $\{x^{(r)}\}$ eran muestras *independientes* de la distribución deseada, los métodos de Monte Carlo con Cadenas de Markov involucran un proceso de Markov en el cual se genera una secuencia de estados $\{x^{(t)}\}$, cada muestra $x^{(t)}$ tiene una distribución de probabilidades que depende del valor previo, $x^{(t-1)}$. Como las muestras sucesivas están correladas unas con otras, la cadena de Markov puede tener que ser simulada durante un tiempo considerable con objeto de generar muestras que sean efectivamente muestras independientes de $P$.

Al igual que era difícil estimar la varianza de un estimador obtenido con muetreo por importancia, también es difícil valorar si un método de Monte Carlo con Cadenas de Markov ha 'convergido', y cuantificar cuánto tiempo tenemos que esperar para obtener muestras que sean efectivamente muestras independientes de $P$.

### 9.4.2. Demostración del método de Metropolis

El método de Metropolis es ampliamente usado para problemas de alta dimensión. Muchas implementaciones del método de Metropolis emplean una distribución propuesta

con una escala de longitud $\epsilon$ que es corta en relación a $L$, la mayor escala de longitud de la región probable. Una razón para elegir una escala de longitud pequeña es que, para muchos problemas de alta dimensión, es muy probable que un paso aleatóreo grande desde un punto típico (esto es, una muestra de $P(\boldsymbol{x})$) termine en un estado que tiene una probabilidad muy baja; por lo que es improbable que esos pasos sean aceptados. Si $\epsilon$ es grande, el movimiento por el espacio de estados sólo ocurrirá cuando sea aceptada esa transición a un estado de baja probabilidad, o cuando un paso aleatóreo grande tenga la suerta de caer en otro estado probable. Por tanto, la tasa de progreso será lenta si se usan pasos grandes.

Por otra parte, la desventaja de los pasos pequeños es que el método de Metropolis explorará la distribución de probabilidad mediante un *paseo aleatóreo,* y un paseo aleatóreo tarda un largo tiempo en llegar a algún sitio, especialmente si damos el paseo con pasos pequeños.

Consideremos un paseo aleatóreo unidimensional, en cada paso del cual nos movemos aleatóreamente a la izquierda o a la derecha con igual probabilidad. Se puede demostrar que después de $T$ pasos de tamaño $\epsilon$, es probable que el estado se haya movido sólo una distancia de $\sqrt{T}\epsilon$.

Recordemos que el primer propósito del muestreo de Monte Carlo es generar varias muestras *independientes* de la distribución dada (una docena, por ejemplo). Si la mayor escala de longitud del espacio de estados es $L$, entonces tenemos que simular el método de Metropolis de paseo aleatóreo durante un tiempo de $T \approx (L/\epsilon)^2$ antes de que podamos obtener una muestra que sea aproximadamente independiente de la condición inicial — y esto es asumiendo que se aceptan todos los pasos: si solamente se acepta de media una fracción $f$ de los pasos, entonces este tiempo se incrementa por un factor de $1/f$.

**Regla empírica: cota inferior del número de iteraciones de un método de Metropolis.** Si la mayor escala de longitud del espacio de estados probables es $L$, un método de Metropolis cuya distribución propuesta genera un paseo aleatóreo con tamaño de paso $\epsilon$ debe ejecutarse durante al menos

$$T \approx (L/\epsilon)^2 \tag{9.31}$$

iteraciones para obtener una muestra independiente.

Esta regla empírica ofrece sólo una cota inferior; la situación puede ser mucho peor, si, por ejemplo, la distribución de probabilidades consiste en varias islas de alta probabilidad separadas por regiones de baja probabilidad.

### 9.4.3. Métodos de Metropolis en altas dimensiones

La regla empírica (ecuación (9.31)), que da una cota inferior para el número de iteraciones de un método de Metropolis de paseo aleatóreo, también se aplica en problemas de más alta dimensión. Considere el caso simple de una distribución objetivo que es una

gaussiana $N$-dimensional, y una distribución propuesta que es una gaussiana esférica de desviación típica $\epsilon$ en cada dirección. Sin pérdida de generalidad, podemos asumir que la densidad objetivo es una distribución separable alineada con los ejes $\{x_n\}$, y que tiene desviación típica $\sigma_n$ en la dirección $n$. Sean $\sigma^{max}$ y $\sigma^{min}$, respectivamente, la mayor y la menor de estas desviaciones típicas. Asumamos que $\epsilon$ se ajusta tal que la frecuencia de aceptación es próxima a 1. Bajo esta asunción, cada variable $x_n$ evoluciona independientemente del resto, ejecutando un paseo aleatóreo con tamaño de paso aproximadamente $\epsilon$. El tiempo requerido para generar muestras de la distribución objetivo efectivamente independientes, será controlado por la mayor escala de longitud $\sigma^{max}$. Al igual que en la sección previa, donde necesitábamos al menos $T \approx (L/\epsilon)^2$ iteraciones para obtener una muestra independiente, aquí necesitamos $T \approx (\sigma^{max}/\epsilon)^2$.

Una cuestión que se plantea inmediatamente es cómo de grande puede ser $\epsilon$. Cuanto más grande sea, menor será el número $T$ de iteraciones, pero si $\epsilon$ es demasiado grande — mayor que $\sigma^{min}$ — entonces la tasa de aceptación caerá abruptamente. Parece admisible que el $\epsilon$ óptimo debe ser similar a $\sigma^{min}$. Estrictamente, esto puede no ser verdad; en casos especiales donde el segundo menor $\sigma_n$ sea significativamente mayor que $\sigma^{min}$, el $\epsilon$ óptimo puede estar muy cerca de ese segundo menor $\sigma_n$. Pero nuestra conclusión aproximada es ésta: donde se usen distribuciones propuestas simples, como las esféricas, necesitaremos al menos $T \approx (\sigma^{max}/\sigma^{min})^2$ iteraciones para obtener una muestra independiente, donde $\sigma^{max}$ y $\sigma^{min}$ son, respectivamente, la mayor y la menor escalas de longitud de la distribución objetivo.

Éstas son buenas y malas noticias. Son buenas noticias porque, a diferencia de los casos de muestreo por importancia y muestro por rechazo, no existe una dependencia catastrófica con la dimensionalidad $N$. Nuestro ordenador dará respuestas útiles en un tiempo menor que la edad del universo. Pero son malas noticias al mismo tiempo, puesto que esta dependencia cuadrática con la ratio de escalas de longitud puede todavía forzarnos a hacer simulaciones de muy larga duración.

Afortunadamente, existen métodos para suprimir los paseos aleatóreos en las simulaciones de Monte Carlo, como veremos en la sección 10.

## 9.5. Muestreo de Gibbs

El muestreo por importancia, el muestreo por rechazo y el método de Metropolis se introdujeron usando ejemplos unidimensionales. El muestreo de Gibbs, es un método para muestrear distribuciones sobre al menos dos dimensiones. El muestreo de Gibbs puede verse como un método de Metropolis en el cual se definen una secuencia de distribuciones propuestas $Q$ en términos de las distribuciones *condicionales* de la densidad conjunta $P(\boldsymbol{x})$. Se asume que, aunque $P(\boldsymbol{x})$ es demasiado compleja para extraer muestras de ella directamente, sus distribuciones condicionales $P(x_i|\{x_j\}_{j\neq i})$ son tratables. Para muchos modelos gráficos (pero no todos) estas distribuciones condicionales unidimensionales son fáciles de muestrear. Por ejemplo, si una distribución gaussiana para ciertas variables $\boldsymbol{d}$ tiene una media desconocida $\boldsymbol{m}$, y la distribución a priori de $\boldsymbol{m}$ es gaussiana, entonces la distribución condicional de $\boldsymbol{m}$ dado $\boldsymbol{d}$ es también gaussiana. Distribuciones condicionales que no tenga una forma estándar pueden ser muestreadas mediante *muestreo por rechazo*

*adaptativo* si las distribuciones condicionales satisfacen ciertas propiedades de convexidad.

Vamos a explicar el muestreo de Gibbs para un caso con dos variables $(x_1, x_2) = \boldsymbol{x}$. En cada iteración, se empieza a partir del estado actual $\boldsymbol{x}^{(t)}$, y se muestrea $x_1$ de la densidad condicional $P(x_1|x_2)$, con $x_2$ fijado a $x_2^{(t)}$. Después se toma una muestra de $x_2$ a partir de la densidad condicional $P(x_2|x_1)$, usando el nuevo valor de $x_1$. Esto nos lleva al nuevo estado $\boldsymbol{x}^{(t+1)}$, y completa la iteración.

En el caso general de un sistema con $K$ variables, una sola iteración involucra muestrear un parámetro cada vez:

$$x_1^{(t+1)} \sim P(x_1|x_2^{(t)}, x_3^{(t)}, \ldots, x_K^{(t)}) \tag{9.32}$$

$$x_2^{(t+1)} \sim P(x_2|x_1^{(t+1)}, x_3^{(t)}, \ldots, x_K^{(t)}) \tag{9.33}$$

$$x_3^{(t+1)} \sim P(x_3|x_1^{(t+1)}, x_2^{(t+1)}, x_4^{(t)}, \ldots, x_K^{(t)}), \text{ etc.} \tag{9.34}$$

### 9.5.1. Convergencia del muestreo de Gibbs hacia la densidad objetivo

Como el muestreo de Gibbs es un método de Metropolis, la distribución de probabilidad de $\boldsymbol{x}^{(t)}$ tiende a $P(\boldsymbol{x})$ conforme $t \to \infty$, mientras $P(\boldsymbol{x})$ no tenga propiedades patológicas.

### 9.5.2. Muestreo de Gibbs en altas dimensiones

El muestreo de Gibbs sufre el mismo defecto que los algoritmos simples de Metropolis – el espacio de estados es explorado mediante un lento paseo aleatóreo, salvo que se haya elegido una parametrización fortuita que haga separable a la densidad $P(\boldsymbol{x})$. Si, por ejemplo, dos variables $x_1$ y $x_2$ están fuertemente correladas, teniendo densidades marginales de ancho $L$ y densidades condicionales de ancho $\epsilon$, entonces necesitaremos al menos unas $(L/\epsilon)^2$ iteraciones para generar una muestra independiente de la densidad objetivo.

Sin embargo, el muestreo de Gibbs involucra parámetros no ajustables, así que es una estrategia atractiva cuando queremos poner un modelo a ejecutarse rápidamente.

## 9.6. Terminología para los métodos de Monte Carlo con Cadenas de Markov

Ahora dedicamos unos momentos a esbozar la teoría en la que se basan el método de Metropolis y el muestreo de Gibbs. Denotamos por $p^{(t)}(\boldsymbol{x})$ la distribución de probabilidades del estado de un simulador basado en cadenas de Markov. (Para visualizar esta distribución, imaginemos que ejecutamos en paralelo una colección infinita de simuladores

idénticos.) Nuestro objetivo es encontrar una cadena de Markov tal que, conforme $t \to \infty$, $p^{(t)}(\boldsymbol{x})$ tienda a la distribución deseada $P(\boldsymbol{x})$.

Una *cadena de Markov* puede especificarse mediante una distribución de probabilidades *inicial* $p^{(0)}(\boldsymbol{x})$ y una *probabilidad de transición* $T(\boldsymbol{x'}; \boldsymbol{x})$.

La distribución de probabilidades del estado en la iteración $(t+1)$-ésima de la cadena de Markov, $p^{(t+1)}(\boldsymbol{x})$, está dada por

$$p^{(t+1)}(\boldsymbol{x'}) = \int T(\boldsymbol{x'}; \boldsymbol{x}) p^{(t)}(\boldsymbol{x}) d^N \boldsymbol{x}. \tag{9.35}$$

### 9.6.1. Propiedades requeridas

Cuando diseñamos un método de Monte Carlo con Cadenas de Markov, construimos una cadena con las siguientes propiedades:

1. La distribución deseada $P(\boldsymbol{x})$ es una *distribución invariante* de la cadena. Una distribución $\pi(\boldsymbol{x})$ es una distribución invariante de la probabilidad de transición $T(\boldsymbol{x'}; \boldsymbol{x})$ si

$$\pi(\boldsymbol{x'}) = \int T(\boldsymbol{x'}; \boldsymbol{x}) \pi(\boldsymbol{x}) d^N \boldsymbol{x}. \tag{9.36}$$

Una distribución invariante es una autofunción de la probabilidad de transición, con autovalor unidad.

2. La cadena también debe ser *ergódica*, es decir,

$$p^{(t)}(\boldsymbol{x}) \to \pi(\boldsymbol{x}) \text{ conforme } t \to \infty, \text{ para cualquier } p^{(0)}(\boldsymbol{x}) \tag{9.37}$$

Un par de razones por las cuales una cadena no podría ser ergódica son:

(a) Su matriz de probabilidades de transición[1] podría ser *reducible*, lo que significa que el espacio de estados contiene dos o más subconjuntos que nuncan pueden ser alcanzados el uno desde el otro. Tales cadenas tienen varias distribuciones invariantes; a cuál $p^{(t)}(\boldsymbol{x})$ se tiende conforme $t \to \infty$ depende de la condición inicial $p^{(0)}(\boldsymbol{x})$.

La matriz de probabilidades de transición de esa cadena tiene más de un autovalor unidad.

(b) La cadena podría tener un conjunto *periódico*, lo cual quiere decir que, para algunas condiciones iniciales, $p^{(t)}(\boldsymbol{x})$ no tiende a una distribución invariante, sino que, en lugar de eso, tiende a un ciclo-límite periódico.

Una cadena de Markov simple con esta propiedad es el paseo aleatóreo por el hipercubo $N$-dimensional. La cadena $T$ toma el estado desde una esquina hasta otra esquina adyacente elegida aleatóreamente. La única distribución invariante de esta cadena es la distribución uniforme sobre los $2^N$ estados, pero la cadena no es ergódica; es periódica con periodo dos: si dividimos los estados en estados con paridad par y estados con paridad impar, notamos que cada estado impar está rodeado por estados pares y *vice versa*. Así que si la condición inicial en el instante $t = 0$ es un estado con paridad par, entonces

[1] La entrada $(i, j)$ de esta matriz se obtiene a partir de la probabilidad de transición como $T(\boldsymbol{x}^{(j)}; \boldsymbol{x}^{(i)})$.

en el instante $t = 1$ — y en todos los instantes impares — el estado debe tener paridad impar, y en todos los instantes pares, el estado será de paridad par.

La matriz de probabilidades de transición de tal cadena tiene más de un autovalor con magnitud igual a la unidad. El paseo aleatóreo por el hipercubo, por ejemplo, tiene autovalores iguales a $+1$ y $-1$.

### 9.6.2. Métodos de construcción de cadenas de Markov

A menudo es conveniente construir $T$ mediante *mezcla* o *concatenación* de *transiciones base* $B$ simples, todas las cuales satisfacen

$$P(\boldsymbol{x}') = \int B(\boldsymbol{x}'; \boldsymbol{x}) P(\boldsymbol{x}) d^N \boldsymbol{x}, \tag{9.38}$$

para la densidad deseada $P(\boldsymbol{x})$, es decir, todas las transiciones base tienen a la densidad deseada como distribución invariante. Estas transiciones base no necesitan ser ergódicas individualmente.

$T$ es una *mezcla* de varias transiciones base $B_b(\boldsymbol{x}', \boldsymbol{x})$ si construimos la transición cogiendo una de las transiciones base al azar, y permitiéndole determinar la transición, es decir,

$$T(\boldsymbol{x}', \boldsymbol{x}) = \sum_b p_b B_b(\boldsymbol{x}', \boldsymbol{x}), \tag{9.39}$$

donde $\{p_b\}$ es una distribución de probabilidades sobre las transiciones base.

$T$ es una *concatenación* de dos transiciones base $B_1(\boldsymbol{x}', \boldsymbol{x})$ y $B_2(\boldsymbol{x}', \boldsymbol{x})$ si inicialmente hacemos una transición a un estado intermedio $\boldsymbol{x}''$ usando $B_1$, y después hacemos una transición desde $\boldsymbol{x}''$ a $\boldsymbol{x}'$ usando $B_2$.

$$T(\boldsymbol{x}', \boldsymbol{x}) = \int B_2(\boldsymbol{x}', \boldsymbol{x}'') B_1(\boldsymbol{x}'', \boldsymbol{x}) d^N \boldsymbol{x}''. \tag{9.40}$$

### 9.6.3. Balance detallado

Muchas probabilidades de transición útiles satisfacen la propiedad del *balance detallado*:

$$T(\boldsymbol{x_a}; \boldsymbol{x_b}) P(\boldsymbol{x_b}) = T(\boldsymbol{x_b}; \boldsymbol{x_a}) P(\boldsymbol{x_a}), \text{ para todo } \boldsymbol{x_b} \text{ y } \boldsymbol{x_a}. \tag{9.41}$$

Esta ecuación dice que si tomamos un estado de la densidad objetivo $P$ y hacemos una transición según $T$ hasta otro estado, es igual de probable tomar $\boldsymbol{x_b}$ e ir desde $\boldsymbol{x_b}$ hasta $\boldsymbol{x_a}$ que tomar $\boldsymbol{x_a}$ e ir desde $\boldsymbol{x_a}$ hasta $\boldsymbol{x_b}$. Las cadenas de Markov que satisfacen el balance detallado también se denominan cadenas de Markov *reversibles*. La razón por la cual la propiedad del balance detallado es interesante es que implica invarianza de la distribución $P(\boldsymbol{x})$ bajo la cadena de Markov $T$, que es una condición necesaria para la propiedad clave que queremos de nuestra simulación MCMC — que la distribución de probabilidades de la cadena converja a $P(\boldsymbol{x})$.

Probar que el balance detallado se cumple es a menudo un paso clave cuando estamos demostrando que una simulación de Monte Carlo con cadenas de Markov (MCMC) converge a la distribución deseada. El método de Metropolis satisface el balance detallado,

por ejemplo. Sin embargo, el balance detallado no es una condición esencial, pues veremos que las cadenas de Markov irreversibles pueden ser útiles en la práctica, porque pueden tener diferentes propiedades en lo relativo al paseo aleatóreo.

## 9.7. Muestreo por rodajas

El muestreo por rodajas es un método de Monte Carlo con cadenas de Markov que tiene analogías con el muestreo por rechazo, el muestreo de Gibbs y el método de Metropolis. Puede aplicarse dondequiera que pueda aplicarse el método de Metropolis, esto es, a cualquier sistema donde pueda evaluarse la densidad $P^*(\boldsymbol{x})$ para todo $\boldsymbol{x}$; tiene la ventaja sobre los métodos simples de Metropolis de que es más robusto a la elección de los parámetros, como, por ejemplo, los tamaños del paso. La versión más simple del muestreo por rodajas es similar al muestreo de Gibbs en el sentido de que consiste en transiciones unidimensionales en el espacio de estados; sin embargo, no existen los requerimientos de que las distribuciones condicionales unidimensionales sean fáciles de muestrear, ni de que tengan ciertas propiedades de convexidad como era requerido en el muestreo por rechazo adaptativo. Y el muestreo por rodajas es similar al muestreo por rechazo en que es un método que asintóticamente extrae muestras del volumen bajo la curva descrita por $P^*(\boldsymbol{x})$; pero no es necesaria una función que haga de cota superior.

Describiremos el muestreo por rodajas dando un boceto de un algoritmo de muestreo unidimensional.

### 9.7.1. El esqueleto del muestreo por rodajas

Asumamos que queremos extraer muestras de $P(\boldsymbol{x}) \propto P^*(\boldsymbol{x})$ donde $x$ es un número real. Un algoritmo unidimensional de muestreo por rodajas es un método para construir transiciones desde un punto bidimensional $(x, u)$ ubicado bajo la curva $P^*(x)$ hasta otro punto $(x', u')$ ubicado bajo la misma curva, tal que la distribución de probabilidades de $(x, u)$ tienda a una distribución uniforme sobre el área bajo la curva $P^*(x)$, cualquiera que sea el punto inicial desde el que comencemos — como la distribución uniforme bajo la curva $P^*(x)$ producida mediante el muestreo por rechazo (sección 9.3).

Una sola transición $(x, u) \rightarrow (x', u')$ de un algoritmo unidimensional de muestreo por rodajas tiene los siguientes pasos, de los cuales los pasos 3 y 8 requerirán una elaboración adicional.

1: evaluar $P^*(x)$
2: extraer una coordenada vertical $u' \sim \text{Uniforme}(0, P^*(x))$
3: crear un intervalo horizontal $(x_l, x_r)$ encerrando a $x$
4: bucle {
5: extaer $x' \sim \text{Uniforme}(x_l, x_r)$
6: evaluar $P^*(x')$
7: si $P^*(x') > u'$ salir del bucle 4-9
8: si no, modificar el intervalo $(x_l, x_r)$
9: }

Existen varios métodos para crear el intervalo $(x_l, x_r)$ en la orden 3, y varios métodos para modificarlo en la orden 8. El punto importante es que el método en conjunto debe satisfacer el balance detallado, de manera que la distribución uniforme para $(x, u)$ bajo la curva $P^*(x)$ sea invariante.

### 9.7.2. El método de *apretar el paso* para la orden 3

En el método de *apretar el paso* para crear un intervalo $(x_l, x_r)$ que encierre a $x$, nos desplazamos en pasos de tamaño $w$ hasta que encontremos unos extremos $x_l$ y $x_r$ en los cuales $P^*$ sea menor que $u'$.

3a: extraer $r \sim \text{Uniform}(0, 1)$
3b: $x_l := x - rw$
3c: $x_r := x + (1 - r)w$
3d: mientras $(P^*(x_l) > u')\{x_l := x_l - w\}$
3e: mientras $(P^*(x_r) > u')\{x_r := x_r + w\}$

### 9.7.3. El método de *encoger el intervalo* para la orden 8

Cuando se extrae un punto $x'$ tal que $(x', u')$ se encuentra por encima de la curva $P^*(x)$, encogemos el intervalo de tal forma que uno de los extremos es $x'$, y el punto original $x$ sigue permaneciendo en el intervalo.

8a: si$(x' > x)\{x_r := x'\}$
8b: si no,$\{x_l := x'\}$

### 9.7.4. Propiedades del muestreo por rodajas

Como un método de Metropolis estándar, el muestreo por rodajas se mueve mediante un paseo aleatóreo, pero mientras que en el método de Metropolis, la elección del tamaño del paso es crítica para la tasa de progreso, en el muestreo por rodajas el tamaño del paso es autosintonizable. Si el tamaño inicial $w$ del intervalo es menor en un factor $f$ que el ancho de la región probable, entonces el procedimiento de *apretar el paso* expandirá el tamaño del intervalo. El coste de apretar el paso es solamente lineal con $f$, mientras que en el método de Metropolis el tiempo de cómputo se escala con el cuadrado de $f$ si el tamaño del paso es demasiado pequeño.

Si el valor $w$ elegido es mayor en un factor $F$ que el ancho de la región probable, entonces el algoritmo gasta un tiempo proporcional al logaritmo de $F$ encogiendo el intervalo hacia el tamaño correcto, ya que el intervalo se encoge típicamente en un factor aproximado de 0.6 cada vez que se rechaza un punto. En cambio, el método de Metropolis responde a un paso demasiado grande rechazando casi todas las propuestas, así que la tasa de progreso decae exponencialmente con $F$. En el muestreo por rodajas no hay rechazos. La probabilidad de permanecer exactamente en el mismo lugar es muy baja.

### 9.7.5. Cómo se usa el muestreo por rodajas en problemas reales

Una densidad $N$-dimensional $P(\boldsymbol{x}) \propto P^*(\boldsymbol{x})$ puede muestrearse con la ayuda del método unidimensional de muestreo por rodajas presentado previamente tomando una secuencia de direcciones $\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \ldots$ y definiendo $\boldsymbol{x} = \boldsymbol{x}^{(t)} + x\boldsymbol{y}^{(t)}$. La función $P(x)$ de la sección 9.7.1 se remplaza por $P^*(\boldsymbol{x}) = P^*(\boldsymbol{x}^{(t)} + x\boldsymbol{y}^{(t)})$. Las direcciones pueden elegirse de distintas formas; por ejemplo, como en el muestreo de Gibbs, las direcciones podrían ser los ejes coordenados; alternativamente, las direcciones $\boldsymbol{y}^{(t)}$ pueden seleccionarse al azar en cualquier manera tal que el procedimiento completo satisfaga el balance detallado.

## 9.8. Cuestiones prácticas adicionales

### 9.8.1. El método de Metropolis para modelos grandes

Nuestra descripción general del método de Metropolis involucraba una actualización conjunta de todas las variables usando una densidad propuesta $Q(\boldsymbol{x'}, \boldsymbol{x})$. Para problemas grandes puede ser más eficiente usar varias distribuciones propuestas $Q^{(b)}(\boldsymbol{x'}, \boldsymbol{x})$, cada una de las cuales actualiza sólo algunas de las componentes de $\boldsymbol{x}$. Cada propuesta se acepta o se rechaza individualmente, y las distribuciones propuestas se aplican repetidamente siguiendo una secuencia.

En el método de Metropolis, la densidad propuesta $Q(\boldsymbol{x'}, \boldsymbol{x})$ suele tener varios parámetros que controlan, por ejemplo, su ancho. Estos parámetros se establecen habitualmente mediante prueba y error con la regla empírica para obtener una frecuencia de rechazo de aproximadamente 0.5. No es válido actualizar los parámetros del ancho dinámicamente durante la simulación en alguna forma que dependa de la historia de la simulación. Tal modificación de la densidad propuesta violaría la condición del balance detallado, que garantiza que la cadena de Markov tiene la distribución invariante correcta.

### 9.8.2. El muestreo de Gibbs en modelos grandes

Nuestra descripción del muestreo de Gibbs involucraba muestrear un único parámetro cada vez, como se describe en las ecuaciones (9.32-9.34). Para problemas grandes puede ser más eficiente muestrear *grupos* de variables conjuntamente,

$$x_1^{(t+1)}, \ldots, x_a^{(t+1)} \sim P(x_1, \ldots, x_a | x_{a+1}^{(t)}, \ldots, x_K^{(t)}) \tag{9.42}$$

$$x_{a+1}^{(t+1)}, \ldots, x_b^{(t+1)} \sim P(x_{a+1}, \ldots, x_b | x_1^{(t+1)}, \ldots, x_a^{(t+1)}, x_{b+1}^{(t)}, \ldots, x_K^{(t)}), \text{ etc.} \tag{9.43}$$

### 9.8.3. ¿Cuántas muestras se necesitan?

Al comienzo de la sección 9, observamos que la varianza de un estimador $\hat{\Phi}$ depende sólo del número de muestras independientes $R$ y del valor de

$$\sigma^2 = \int P(\boldsymbol{x})(\phi(\boldsymbol{x}) - \Phi)^2 d^N\boldsymbol{x}. \tag{9.44}$$

Hasta ahora hemos discutido diferentes métodos de generar muestras de $P(\boldsymbol{x})$. ¿Cuántas muestras independientes deberíamos aspirar a conseguir?

En muchos problemas, realmente sólo necesitamos aproximadamente doce muestras de $P(\boldsymbol{x})$. Imagine que $\boldsymbol{x}$ es un vector desconocido que representa la cantidad de corrosión presente en cada una de las 10000 tuberías de Cambridge, y $\phi(\boldsymbol{x})$ es el coste total de reparar esas tuberías. La distribución $P(\boldsymbol{x})$ describe la probabilidad de un estado $\boldsymbol{x}$ dados los tests que han sido efectuados en algunas tuberías y las asunciones sobre la física de la corrosión. La cantidad $\sigma^2$ es la varianza del coste – $\sigma$ mide cuánto deberíamos suponer que el coste actual difiera del coste esperado $\Phi$. Ahora nos preguntamos con qué precisión le gustaría a un gestor conocer $\Phi$. Existe poco interés en conocer $\Phi$ con una precisión mejor que $\sigma/3$. Después de todo, el coste real es probable que difiera en una cantidad $\pm\sigma$ de $\Phi$. Si obtenemos $R = 12$ muestras independientes de $P(\boldsymbol{x})$, podemos estimar $\Phi$ con una precisión de $\sigma/\sqrt{12}$ – que es inferior a $\sigma/3$. Por tanto, doce muestras bastan.

### 9.8.4. Asignación de recursos

Asumiendo que ya hemos decidido cuántas muestras independientes $R$ se requieren, una cuestión importante es cómo hacer uso de los recursos limitados de nuestro ordenador para obtener estas muestras.

Un experimento típico de Monte Carlo con cadenas de Markov involucra un periodo inicial en el cual se pueden ajustar los parámetros de control de la simulación, tales como tamaños del paso. Éste es seguido por un periodo de *burn in* durante el cual esperamos que la simulación converja a la distribución deseada. Finalmente, conforme la simulación continúa, grabamos el vector de estados de vez en cuando para crear una lista de estados $\{\boldsymbol{x}^{(r)}\}_{r=1}^{R}$ que esperamos que sean muestras aproximadamente independientes de $P(\boldsymbol{x})$.

Existen distintas estrategias posibles:

1. Hacer una ejecución larga, obteniendo $R$ muestras de ella.

2. Hacer unas pocas ejecuciones de longitud media con diferentes condiciones iniciales, obteniendo algunas muestras de cada una.

3. Hacer $R$ ejecuciones cortas, cada una empezando desde una condición inicial diferente elegida al azar. El único estado que grabamos es el estado final de cada simulación.

La primera estrategia tiene las mejores posibilidades de lograr la convergencia. La última estrategia puede tener la ventaja de que las correlaciones entre las muestras grabadas son pequeñas. El camino intermedio es popular entre los expertos de MCMC porque evita la ineficiencia de descartar iteraciones *burn in* en muchas ejecuciones, mientras que sigue permitiendo detectar problemas de ausencia de convergencia que no serían aparentes con una sola ejecución.

Finalmente, se debería enfatizar que no existe necesidad de hacer que las muestras sean independientes. Promediar sobre puntos dependientes es correcto — no conducirá a ningún sesgo en la estimación. Por ejemplo, cuando usamos la estrategia 1 ó 2, podemos, si lo deseamos, incluir todos los puntos entre la primera y la última muestra de cada ejecución. Por supuesto, estimar la precisión del estimador es más difícil cuando los puntos son dependientes.

## 9.9. Resumen

- Los métodos de Monte Carlo son una poderosa herramienta que permite muestrear cualquier distribución de probabilidades que pueda expresarse de la forma $P(\boldsymbol{x}) = \frac{1}{Z}P^*(\boldsymbol{x})$.
- Los métodos de Monte Carlo pueden responder virtualmente cualquier cuestión relacionada con $P(\boldsymbol{x})$ poniendo la cuestión en la forma

$$\int \phi(\boldsymbol{x})P(\boldsymbol{x})d^N\boldsymbol{x} \approx \frac{1}{R}\sum_r \phi(x^{(r)}). \tag{9.45}$$

- En problemas de alta dimensión, los únicos métodos satisfactorios son aquéllos basados en cadenas de Markov, como el método de Metropolis, el muestreo de Gibbs y el muestreo por rodajas. El muestreo de Gibbs es un método atractivo porque no tiene parámetros ajustables, pero su uso se restringe a casos donde puedan generarse muestras de las distribuciones condicionales. El muestreo por rodajas es atractivo porque, aunque tiene parámetros de longitud del paso, su rendimiento no es muy sensible a los valores de esos parámetros.
- Los algoritmos de Metropolis y los algoritmos de muestreo de Gibbs, aunque ampliamente usados, ofrecen prestaciones pobres porque exploran el espacio mediante un lento paseo aleatóreo. En la sección 10 discutiremos un método que se utiliza para acelerar las simulaciones MCMC.
- El muestreo por rodajas no evita el comportamiento de paseo aleatóreo, pero elige automáticamente el mayor paso apropiado, reduciendo, por tanto, los efectos negativos del paseo aleatóreo en comparación con, por ejemplo, un método de Metropolis con un paso pequeñito.