Tema 8: Métodos de cadenas de Markov Monte Carlo

Conchi Ausín
Departamento de Estadística
Universidad Carlos III de Madrid
concepcion.ausin@uc3m.es

CESGA, Noviembre 2012

Introducción

- Los métodos de cadenas de Markov Monte Carlo (MCMC) son métodos de simulación para generar muestras de las distribuciones a posteriori y estimar cantidades de interés a posteriori.
- En los métodos MCMC se simulan valores sucesivamente de una densidad propuesta, que no tiene que ser necesariamente parecida a la densidad a posteriori.
- Cada valor generado depende sólo del anterior valor simulado, de ahí la noción de cadena de Markov.
- Los métodos MCMC no son exclusivos de la inferencia Bayesiana, sino que pueden usarse para simular valores de una distribución de la que no es fácil generar muestras.
- Sin embargo, estos métodos son muy usados en la computación moderna Bayesiana ya que en la mayoría de los casos la forma analítica de $\pi(\theta \mid \mathbf{x})$ es deconocida y la dimensión de θ elevada.

Contenidos

Cadenas de Markov

Una cadena de Markov es una secuencia de variables, $\Theta_0, \Theta_1, \Theta_2, \ldots$ tal que la distribucion de Θ_t dados los valores previos Θ_0, Θ_1, $\Theta_2, \ldots, \Theta_{t-1}$ sólo depende de Θ_{t-1},

$$\Pr\left(\Theta_t \in A \mid \Theta_0 = \theta_0, \Theta_1 = \theta_1, ..., \Theta_{t-1} = \theta_{t-1}\right) = \Pr\left(\Theta_t \in A \mid \Theta_{t-1} = \theta_{t-1}\right)$$

Una cadena de Markov (homogénea en el tiempo) viene definida por su estado inicial, θ_0, y el núcleo de transición:

$$\Pr(\Theta_{t+1} = \theta_2 | \Theta_t = \theta_1).$$

Bajo ciertas condiciones, una cadena de Markov converge a su distribución estacionaria:

$$\lim_{t \to \infty} \Pr(\Theta_t = \theta | \Theta_0 = \theta_0) = \pi(\theta)$$

Ideas básicas sobre los métodos MCMC

Supongamos que queremos simular valores de una distribución a posteriori $\pi(\theta \mid \mathbf{x})$.

La idea de los métodos MCMC consiste en simular una cadena de Markov, $\theta_1, \theta_2 \ldots$, cuya distribución estacionaria sea $\pi(\theta \mid \mathbf{x})$.

Cada valor simulado, θ_t, depende únicamente de su predecesor, θ_{t-1}.

Si el algoritmo se implementa correctamente, la convergencia de la cadena está garantizada independentemente de cuáles sean los valores iniciales.

Es necesario simular la cadena para un número elevado de iteraciones para aproximarse a la distribución estacionaria.

Los primeros valores simulados, (iteraciones de *burn-in*), se eliminan porque no están en el estado estacionario.

Algoritmo Metropolis Hastings

El algoritmo MH simula de una cadena de Markov cuya distribución estacionaria es $\pi(\theta \mid \mathbf{x})$. Se comienza con un valor inicial θ_0.

1. Dado el valor actual, θ_t, simular un valor candidato $\tilde{\theta}$, de una densidad propuesta, $q(\tilde{\theta} \mid \theta_t)$.
2. Calcular la probabilidad de aceptar el valor generado:

$$\alpha = \text{mín}\left\{1, \frac{\pi\left(\tilde{\theta} \mid \mathbf{x}\right) q\left(\theta_t \mid \tilde{\theta}\right)}{\pi\left(\theta_t \mid \mathbf{x}\right) q\left(\tilde{\theta} \mid \theta_t\right)}\right\}.$$

3. Simular u de una distribución uniforme $\mathcal{U}(0,1)$.
4. Si $u < \alpha$, tomar $\theta_{t+1} = \tilde{\theta}$. Si no, rechazar y tomar $\theta_{t+1} = \theta_t$.
5. Volver a 1.

Algoritmo Metropolis Hastings

La probabilidad de aceptar, α, no depende de la constante de integración de la distribución a posteriori, de modo que,

$$\alpha = \text{mín}\left\{1, \frac{\pi\left(\tilde{\theta}\right) f\left(\mathbf{x} \mid \tilde{\theta}\right) q\left(\theta_t \mid \tilde{\theta}\right)}{\pi\left(\theta_t\right) f\left(\mathbf{x} \mid \theta_t\right) q\left(\tilde{\theta} \mid \theta_t\right)}\right\}.$$

- Un caso particular es el algoritmo Metropolis, en el que la densidad propuesta $q(\tilde{\theta} \mid \theta)$ es simétrica y la probabilidad de aceptación se reduce a:

$$\alpha = \text{mín}\left\{1, \frac{\pi\left(\tilde{\theta}\right) f\left(\mathbf{x} \mid \tilde{\theta}\right)}{\pi(\theta_t) f\left(\mathbf{x} \mid \theta_t\right)}\right\}.$$

  Si además, $q\left(\tilde{\theta} \mid \theta\right) = q\left(\left|\tilde{\theta} - \theta\right|\right)$, se llama algoritmo metropolis de paseo aleatorio o algoritmo RWMH.

Algoritmo Metropolis Hastings

El algoritmo MH no es más eficiente si α es muy elevada ya que la autocorrelación entre los valores será muy alta y la cadena se moverá lentamente por la distribución a posteriori de θ.

- Por ejemplo, en el algoritmo RWMH, la tasa de valores aceptados, α, debe rondar el 25 % si el número de parámetros es elevado. Para modelos con 1 ó 2 parámetros, α debe rondar el 50 %. Una elección habitual en el caso univariante es usar:

$$q\left(\tilde{\theta} \mid \theta_t\right) = \mathcal{N}(\theta_t, \sigma^2),$$

donde el valor de σ se ajusta para obtener un valor aceptable de α.

Algoritmo Metropolis Hastings

- Otro caso particular es el algoritmo MH de independencia en el que

$$q\left(\tilde{\theta} \mid \theta_t\right) = q\left(\tilde{\theta}\right)$$

  independendiente de θ_t.

- Este algoritmo funciona bien si la densidad de $q(\theta)$ es similar a $\pi(\theta \mid \mathbf{x})$, aunque con cola más pesadas, como en el método de rechazo.

- En este caso, la tasa α es mejor cuanto más elevada.

Algoritmo Metropolis Hastings

Ejemplo 8.1. Considerar una muestra de tamaño n de una distribución Cauchy, $X \mid \theta \sim \mathcal{C}(\theta, 1)$. Asumiendo una distribución a priori impropia $\pi(\theta) \propto 1$, obtener una muestra de la distribución a posteriori:

1. Usando un algoritmo RWMH con una densidad propuesta Cauchy, $\tilde{\theta} \mid \theta \sim \mathcal{C}(\theta_t, \sigma)$, ajustando el parámetro de escala, σ, para obtener una tasa razonable de valores aceptados.
2. Usando un algoritmo MH de independencia con una densidad propuesta Cauchy, $\tilde{\theta} \sim \mathcal{C}(m, \tau)$, ajustando el parámetro de escala, τ, para obtener una tasa razonable de valores aceptados.

La función de densidad de una distribución Cauchy, $\theta \sim \mathcal{C}(m, \tau)$ es:

$$f(\theta \mid \mu, \sigma) = \frac{1}{\pi\sigma\left(1 + \left(\frac{\theta-\mu}{\sigma}\right)^2\right)}.$$

Algoritmo Metropolis Hastings

Cuando la dimensión de θ es alta, es difícil encontrar una densidad propuesta. En este caso, se puede dividir en bloques, $\theta = (\theta_1, \ldots, \theta_k)$, y definir un algoritmo MH por bloques.

Supongamos que se divide $\theta = (\theta_1, \theta_2, \theta_3)$ y se definen tres densidades propuestas, q_1, q_2 y q_3. Partiendo de $(\theta_{1,0}, \theta_{2,0}, \theta_{3,0})$, se repite:

1. Simular $\tilde{\theta}_1 \sim q_1\left(\tilde{\theta}_1 \mid \theta_{2,t}, \theta_{3,t}\right)$ y se acepta con probabilidad:

$$\alpha = \text{mín}\left\{1, \frac{\pi\left(\tilde{\theta}_1 \mid \mathbf{x}, \theta_{2,t}, \theta_{3,t}\right) q_1\left(\theta_{1,t} \mid \tilde{\theta}_1, \theta_{2,t}, \theta_{3,t}\right)}{\pi\left(\theta_{1,t} \mid \mathbf{x}, \theta_{2,t}, \theta_{3,t}\right) q_1\left(\tilde{\theta}_1 \mid \theta_{1,t}, \theta_{2,t}, \theta_{3,t}\right)}\right\}$$

2. Simular $\tilde{\theta}_2 \sim q_2\left(\tilde{\theta}_2 \mid \theta_{1,t+1}, \theta_{3,t}\right)$ y aceptar con análoga probabilidad.
3. Simular $\tilde{\theta}_3 \sim q_3\left(\tilde{\theta}_3 \mid \theta_{1,t+1}, \theta_{2,t+1}\right)$ y aceptar con análoga probabilidad.

Muestreo de Gibbs

El muestreo de Gibbs es un caso particular del algoritmo MH por bloques en cual las densidades propuestas coinciden con las distribuciones a posteriori condicionadas de modo que la probabilidad de aceptar es siempre uno.

Supongamos que el conjunto de parámetros es $\theta = (\theta_1, \ldots, \theta_k)$ y que para todo $i = 1, \ldots, k$, es fácil simular de la distribución a posteriori condicional, $\pi(\theta_i \mid \theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_k)$

Fijar un valor inicial, $(\theta_{1,0}, \ldots, \theta_{k,0})$. Repetir:

1. Simular $\theta_{1,t+1} \sim \pi(\theta_1 \mid \theta_{2,t}, \ldots, \theta_{k,t})$
2. Simular $\theta_{2,t+1} \sim \pi(\theta_2 \mid \theta_{1,t+1}, \theta_{3,t} \ldots, \theta_{k,t})$
3. Simular $\theta_{3,t+1} \sim \pi(\theta_3 \mid \theta_{1,t+1}, \theta_{2,t+1}, \theta_{4,t}, \ldots, \theta_{k,t})$
4. $\vdots$
5. Simular $\theta_{k,t+1} \sim \pi(\theta_k \mid \theta_{1,t+1}, \ldots, \theta_{k-1,t+1})$

Muestreo de Gibbs

Ejemplo 8.2. La distribución normal-gamma a priori para los parámetros (μ, τ) de una densidad normal puede no ser una elección a priori adecuada ya que la distribución de μ depende a priori de ϕ.

Una alternativa es asumir que μ y τ son independientes a priori tal que:

$$\mu \sim \mathcal{N}\left(m, \frac{1}{\psi}\right), \qquad \phi \sim \mathcal{G}\left(\frac{a}{2}, \frac{b}{2}\right)$$

donde m, ψ, a, b son fijos. Entonces, es fácil obtener que las distribuciones a posterior condicionales son:

$$\mu \mid \phi, \mathbf{x} \sim N\left(\frac{n\phi\bar{x} + \psi m}{n\phi + \psi}, \frac{1}{n\phi + \psi}\right)$$

$$\phi \mid \mu, \mathbf{x} \sim G\left(\frac{a+n}{2}, \frac{b + (n-1)\,s^2 + n\,(\mu - \bar{x})^2}{2}\right)$$

Simular mediante un muestreo de Gibbs valores de $\pi(\mu, \phi \mid \mathbf{x})$ para los datos de las longitudes del caparazón en cangrejos del tema 6.

Muestreo de Gibbs

Dentro de un algoritmo MCMC, se pueden combinar pasos de tipo Gibbs sampling con pasos MH, dependiendo de si la distribución a posteriori condicional de los parámetros es o no fácil de simular.

Ejemplo 8.3. Considerar una muestra de tamaño n de una distribución gamma, $X \mid \nu, \lambda \sim \mathcal{G}(\nu, \lambda)$ con función de densidad:

$$f(x \mid \nu, \lambda) = \frac{\lambda^{\nu}}{\Gamma(\nu)} x^{\nu-1} \exp(-\lambda x), \qquad X > 0.$$

Asumiendo a priori:

$$\nu \sim \mathcal{U}(0, 100), \qquad \lambda \sim \mathcal{G}(a, b),$$

construir un algoritmo MCMC para simular de la distribución a posteriori de los parámetros.

Muestreo de Gibbs

Comentarios

- Cuando las distribuciones a posteriori condicionales no son fáciles de simular, se pueden combinar también con métodos de simulación Monte Carlo directa del tipo aceptación-rechazo.
- El software `Winbugs` se basa en construcciones de algoritmos de tipo Gibbs sampling en la que las distribuciones a posteriori condicionales se simulan mediante refinamientos del método de rechazo, conocidos como algoritmos ARS y ARMS.
- Estos métodos proporcionan métodos de simulación automáticos en los que no es necesario calibrar la tasa de valores aceptados.
- La desventaja es que a menudo proporcionan métodos menos eficientes que los algoritmos construidos específicamente para un problema en concreto.

Diagnosis de convergencia

Cuando se ejecuta un algoritmo MCMC, es importante examinar si los valores simulados, θ_t, han convergido aproximadamente a la distribución estacionaria, $\pi(\theta \mid \mathbf{x})$.

Para ello es recomendable:

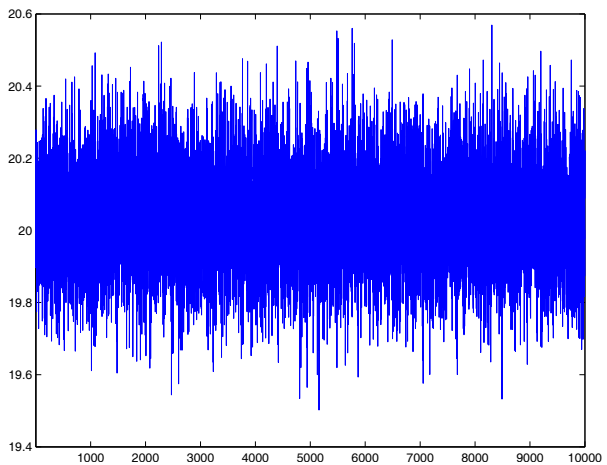
1. Examinar cómo de bien esté explorando el algoritmo MCMC el espacio de estados..
2. Verificar la convergencia de las medias de los valores simulados en el MCMC, e.g. $\frac{1}{N}\sum_{t=1}^{N}\theta_t \rightarrow E[\theta \mid \mathbf{x}]$.
3. Analizar si los valores simulados son aproximadamente una muestra de valores independientes e idénticamente distribuidos.

Existen además numerosos procedimientos en la literatura para estudiar la convergencia de la cadena. Una posibilidad es ejecutar el algoritmo varias veces comenzando en distintos valores iniciales y comprobar si el comportamiento de la distribución estacionaria es la misma.

Diagnosis de convergencia

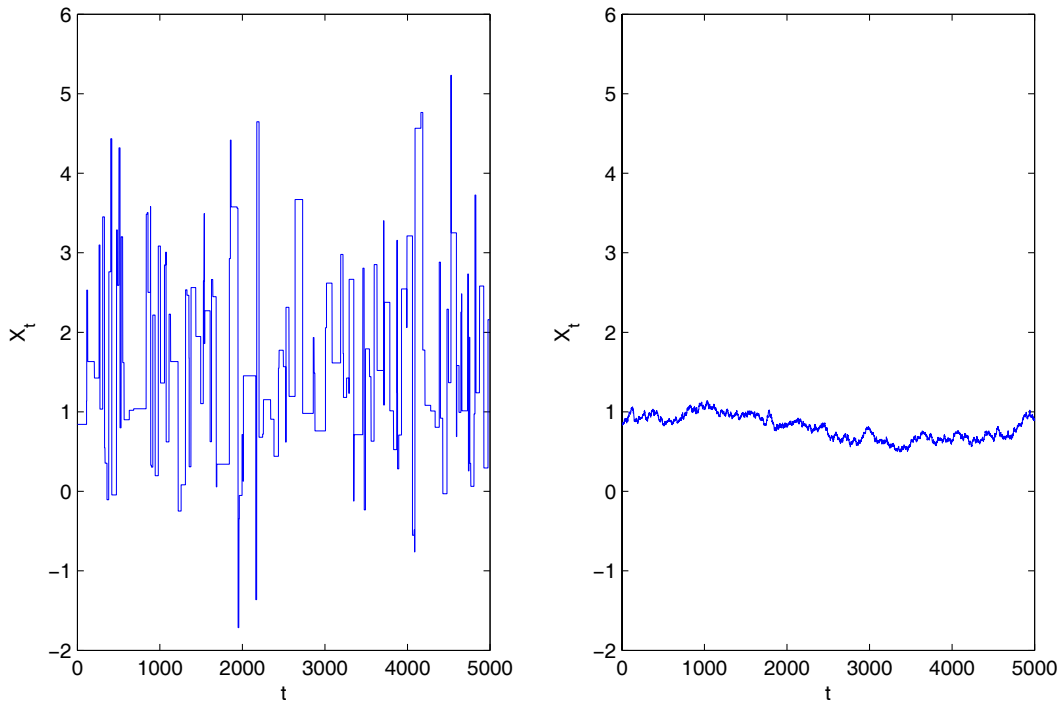
Para examinar si la cadena está explorando bien el espacio de estados, se pueden graficar los valroes simulados de θ_t frente a t y analizar si se producen desviaciones de la estacionaridad.

Por ejemplo, el siguiente gráfico muestra la traza de θ en el ejemplo 8.1. La cadena presenta una buena exploración de estados.



Diagnosis de convergencia

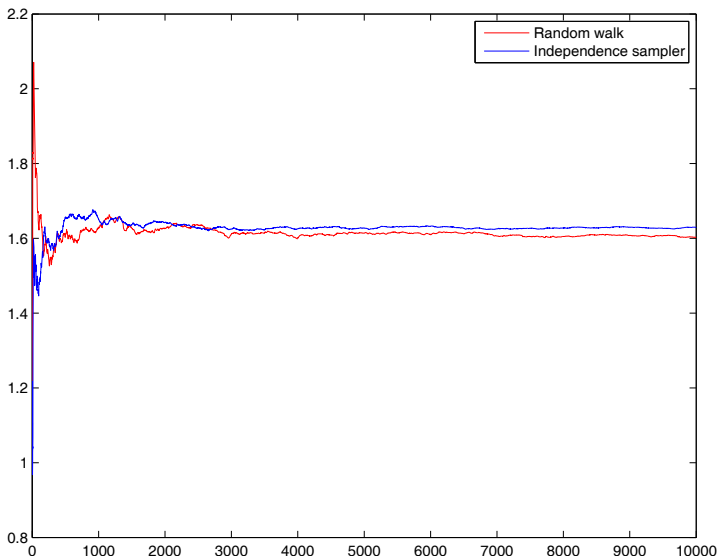
Los siguientes diagramas muestran ejemplos de mal comportamiento en la exploración de estados de la cadena.



Diagnosis de convergencia

Para examinar la convergencia de las medias a posteriori de los parámetros estimados, se pueden graficar el valor de la media muestral de los valores simulados en función de t y verificar si han convergido.

El siguiente gráfico muestra las estimaciones de $E[\theta \mid \mathbf{x}]$ obtenidas para los dos algoritmos del ejemplo 8.1.

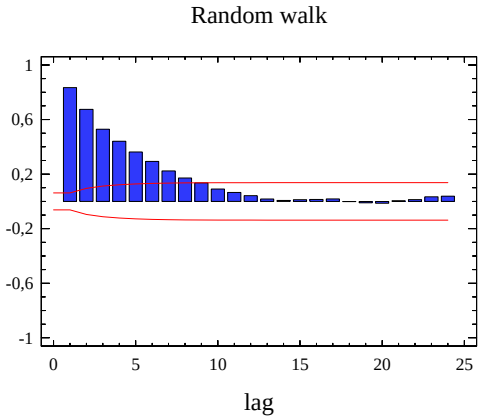


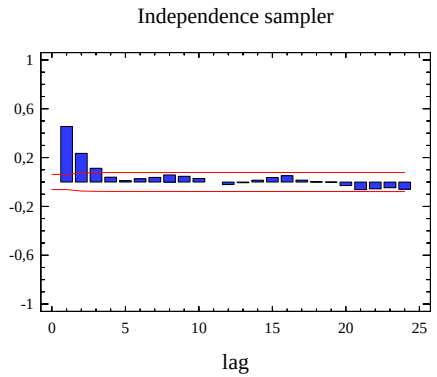
Parece que tras 3000 iteraciones, las medias han convergido, así que una posibilidad es usar 3000 iteraciones de burnin.

Diagnosis de convergencia

Finalmente, se pueden graficar las funciones de autocorrelación de los valores generados. En general, como se simulan valores de una cadena de Markov, los valores de θ_t estáran correlados positivamente.

El siguiente gr'afico muestra los gráficos de autocorrelación del parámetro θ oobtenidos en los dos algoritmos del ejemplo 8.1.





En el algoritmo RWMH, la autocorrelación desaparece después del período 9 y en el muestreo de independencia sobre el período 4. Una posibilidad es guardar los valores simulados cada 9 y 4 iteraciones, respectivamente, para tener una muestra de valores aproximadamente indeppendientes.