CAPÍTULO 10. MÉTODOS MCMC

Para leer

Gelman et al (1995) Capítulo 11,

Lee (1997), Capítulo 9, Secciones 9.4–9.5,

Gilks et al (1996) contiene muchas aplicaciones.

Sitios útiles del red

MCMC preprint service:
`www.statslab.cam.ac.uk/~mcmc/`

Bugs software:
`www.mrc-bsu.cam.ac.uk/bugs/welcome.shtml`

La idea básica: cadenas de Markov

Definición 14 *Un proceso estocástico X_t es una cadena de Markov si*

$$f(x_t|x_1,\ldots,x_{t-1}) = f(x_t|x_{t-1}).$$

Bajo ciertas condiciones ((aperiodicidad, irreducibilidad), recurrencia Harris), la distribución estacionaria $f(x)$ de la cadena es única y cumple

$$f(x) = \int f(y)P(X_t = x|X_{t-1} = y)\, dy.$$

Supongamos que existe una cadena de Markov cuya distribución estacionaria es $f(\boldsymbol{\theta}|\mathbf{x})$. Entonces, se puede muestrear la cadena de Markov y después de un tiempo suficiente, los datos muestreados simulan una muestra de la distribución a posteriori.

El algoritmo Metropolis-Hastings

Este algorítmo (Metropolis et al 1953, Hastings 1970) es el algoritmo más general. Para una buena introducción ver Chib y Greenberg (1995).

Sea la densidad a posteriori

$$f(\boldsymbol{\theta}|\mathbf{x}) \propto l(\boldsymbol{\theta}|\mathbf{x}) f(\boldsymbol{\theta}).$$

Se define una densidad condicional $g\left(\boldsymbol{\theta}^*|\boldsymbol{\theta}\right)$ (*proposal distribution*) que es fácil muestrear.

El siguiente algoritmo simula una muestra de una cadena de Markov con distribución estacionaria $f(\boldsymbol{\theta}|\mathbf{x})$.

El Algoritmo

El algoritmo Metropolis-Hastings es:

1. Definir un valor inicial $\boldsymbol{\theta}^{(0)}$,

2. $t = 0$,

3. Generar $\boldsymbol{\phi} \sim g\left(\boldsymbol{\phi}|\boldsymbol{\theta}^{(t)}\right)$,

4. Definir

$$\alpha\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}\right) = \text{mín}\left[1, \frac{f(\boldsymbol{\phi})l(\boldsymbol{\phi}|\mathbf{x})g\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\phi}\right)}{f(\boldsymbol{\theta}^{(t)})l(\boldsymbol{\theta}^{(t)}|\mathbf{x})g\left(\boldsymbol{\phi}|\boldsymbol{\theta}^{(t)}\right)}\right]$$

5. Tomar

$$\boldsymbol{\theta}^{(t+1)} = \begin{cases} \boldsymbol{\phi} & \text{con probabilidad } \alpha \\ \boldsymbol{\theta}^{(t)} & \text{en caso contrario} \end{cases}$$

6. $t = t + 1$. Ir a 3.

Demostración parcial del algoritmo.

Observamos en primer lugar que el algorítmo define una cadena de Markov. Las probabilidades de transición $P\left(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}\right)$ son

$$P\left(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}\right) = g\left(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}\right)\alpha\left(\boldsymbol{\theta}^{(t)},\boldsymbol{\theta}^{(t+1)}\right) + I_{\boldsymbol{\theta}^{(t+1)}=\boldsymbol{\theta}^{(t)}}\left[1-\int g\left(\boldsymbol{\phi}|\boldsymbol{\theta}^{(t)}\right)\alpha\left(\boldsymbol{\theta}^{(t)},\boldsymbol{\phi}\right)\,d\boldsymbol{\phi}\right]$$

y observando que

$$\frac{f(\boldsymbol{\theta}^{(t)}|\mathbf{x})g\left(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}\right)\alpha\left(\boldsymbol{\theta}^{(t)},\boldsymbol{\theta}^{(t+1)}\right)}{f(\boldsymbol{\theta}^{(t+1)}|\mathbf{x})g\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t+1)}\right)\alpha\left(\boldsymbol{\theta}^{(t+1)},\boldsymbol{\theta}^{(t)}\right)} = 1$$

se tiene la siguiente ecuación (*balance equation*)

$$f(\boldsymbol{\theta}^{(t)}|\mathbf{x})P\left(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}\right) = f(\boldsymbol{\theta}^{(t+1)}|\mathbf{x})P\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t+1)}\right)$$

y integrando con respeto a $\boldsymbol{\theta}^{(t)}$ nos lleva

$$\int f(\boldsymbol{\theta}^{(t)}|\mathbf{x})P\left(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}\right)\,d\boldsymbol{\theta}^{(t)} = f(\boldsymbol{\theta}^{(t+1)}|\mathbf{x})$$

Se ha demostrado que la distribución a posteriori es la distribución invariante de la cadena. También se necesita demostrar la existencia del equilíbrio.

Claramente la cadena es aperiodica si

$$g(\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} | \boldsymbol{\theta}^{(t)}) > 0$$

y la cadena es irreducible si

$$g(\boldsymbol{\phi}|\boldsymbol{\theta}) > 0 \; \forall \; \boldsymbol{\phi}$$

Recurrencia Harris significa que la probabilidad de que la cadena vuelva al estado $\boldsymbol{\theta}$ un número infinito de veces es 1. Más o menos, la cadena cumple esta condición si es irreducible. Ver, por ejemplo, Robert y Casella (1999).

Roberts y Smith (1994) proporcionan algunas condiciones más sencillas para la convergencia del algoritmo.

Observación 80 *En teoría, se puede utilizar (casi) cualquier distribución* $g(\boldsymbol{\phi}|\boldsymbol{\theta})$*. Lo más importante es que es fácil muestrearla.*

Existen resultados que demuestran que la tasa de convergencia del algorítmo es (a menudo) por lo menos geométrica. Ver por ejemplo Roberts y Tweedie (1996). No obstante, la convergencia del algoritmo dependerá mucho de la elección de $g(\cdot)$*.*

Si se rechazan demasiados movimientos, la convergencia será muy lenta.

En el caso particular de que $g(\cdot)$ *sea proporcional a la distribución a posteriori, se tiene* $\alpha = 1$ *y el método es igual al método de Monte-Carlo simple.*

El algoritmo con partición

Si $\boldsymbol{\theta}$ es de alta dimensión, la elección de $g(\cdot)$ parece complicada.

En muchos casos es más fácil dividir el vector $\boldsymbol{\theta}$ en bloques $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k)$ y muestrear cada bloque individualmente.

$$\text{Sea } \boldsymbol{\theta}_{-i} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{i-1}, \boldsymbol{\theta}_{i+1}, \ldots, \boldsymbol{\theta}_k).$$

Sea $g_i(\boldsymbol{\phi}_i|\boldsymbol{\theta}) = g_i(\boldsymbol{\phi}_i|\boldsymbol{\theta}_i, \boldsymbol{\theta}_{-i})$ una distribución de propuesta para el bloque i, $(i = 1, \ldots, k)$.

Entonces, se define el siguiente algoritmo Metropolis Hastings por bloques.

Algorítmo

1. $t = 0$, Definir un valor inicial $\boldsymbol{\theta}^{(0)}$,

2. Para $i = 1, \ldots, k$

   a) Generar $\boldsymbol{\phi}_i \sim g_i\left(\boldsymbol{\phi}_i | \boldsymbol{\theta}^{(t)}\right)$,

   b) Definir

$$\alpha_i\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}_i\right) = \text{mín}\left[1, \frac{f(\boldsymbol{\phi}_i|\boldsymbol{\theta}_{-i}^{(t)}) l(\boldsymbol{\phi}_i|\mathbf{x}, \boldsymbol{\theta}_{-i}^{(t)}) g_i\left(\boldsymbol{\theta}_i^{(t)}|\boldsymbol{\phi}, \boldsymbol{\theta}_{-i}^{(t)}\right)}{f(\boldsymbol{\theta}_i^{(t)}|\boldsymbol{\theta}_{-i}^{(t)}) l(\boldsymbol{\theta}_i^{(t)}|\mathbf{x}, \boldsymbol{\theta}_{-i}^{(t)}) g_i\left(\boldsymbol{\phi}_i|\boldsymbol{\theta}_i^{(t)}, \boldsymbol{\theta}_{-i}^{(t)}\right)}\right]$$

   c) Tomar

$$\boldsymbol{\theta}_i^{(t)} = \begin{cases} \boldsymbol{\phi}_i & \text{con probabilidad } \alpha \\ \boldsymbol{\theta}_i^{(t)} & \text{en caso contrario} \end{cases}$$

3. $t = t + 1$. Ir a 2.

El algoritmo tiene la ventaja de que con una separación de los parámetros, se simplifica el cálculo de las probabilidades de aceptación. En muchas situaciones, es bastante fácil evaluar las distribuciones condicionales.

¿Cómo decidir si hay convergencia en la práctica?

Aunque teóricamente la convergencia del algorítmo está garantizada, en la práctica se necesitan usar algunos criterios de decisión. Existen varias posibilidades.

Múltiples simulaciones con valores iniciales dispersos

Ver Gelman y Rubin (1992).

Ejemplo 116 *Se quiere estimar* $E[\theta|\mathbf{x}]$.

Corremos J *cadenas durante* R *iteraciones. Sea* $\overline{\theta}_{\cdot j} = \frac{1}{R}\sum_{i=1}^{R}\theta^{(ij)}$ *donde* $\theta^{(ij)}$ *es el valor* i *generado por la cadena* j. *Sea* $\overline{\theta}_{\cdot\cdot} = \frac{1}{J}\sum_{j=1}^{J}\overline{\theta}_{\cdot j}$.

Se calculan las varianzas (B) entre y (W) dentro de cada series.

$$B = \frac{R}{J-1}\sum_{j=1}^{J}(\overline{\theta}_{\cdot j} - \overline{\theta}_{\cdot\cdot})^2$$

$$W = \frac{1}{J}\sum_{j=1}^{J} s_j^2$$

donde s_j^2 es la varianza estimada en la serie j.

Una estimación de la varianza a posteriori $V[\theta|\mathbf{x}]$ es

$$\widehat{V}^{+} = \frac{J-1}{J}W + \frac{1}{R}B$$

que sobrestima la verdadera varianza si los puntos iniciales son muy dispersos pero es insesgado suponiendo estacionaridad (cuando $J \to \infty$).

No obstante, para J finita, la varianza estimada dentro de cada secuencia, W, debe subestimar

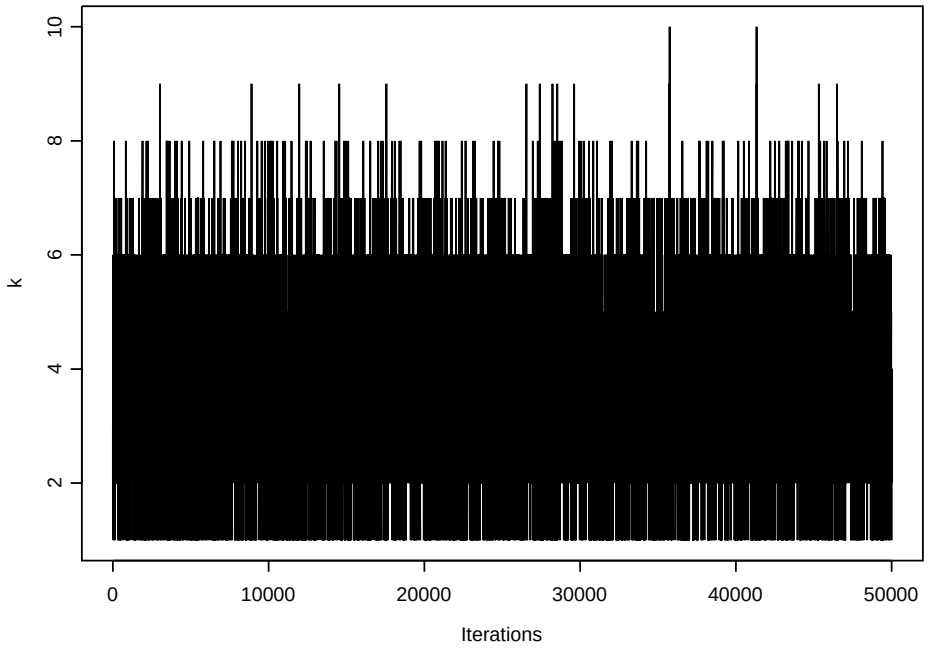
$V[\theta|\mathbf{x}]$ *porque las cadenas no han tenido tiempo para muestrear toda la distribución estacionaria. Pero, cuando* $J \to \infty$, W *también es insesgada.*

Calculando la razón $\frac{\hat{V}^+}{W}$, *si es mucho más grande que* 1, *proporciona evidencia de falta de convergencia.*

El problema con múltiples simulaciones es que tardan más tiempo.

Gráficos de los valores generados $\theta^{(t)}$

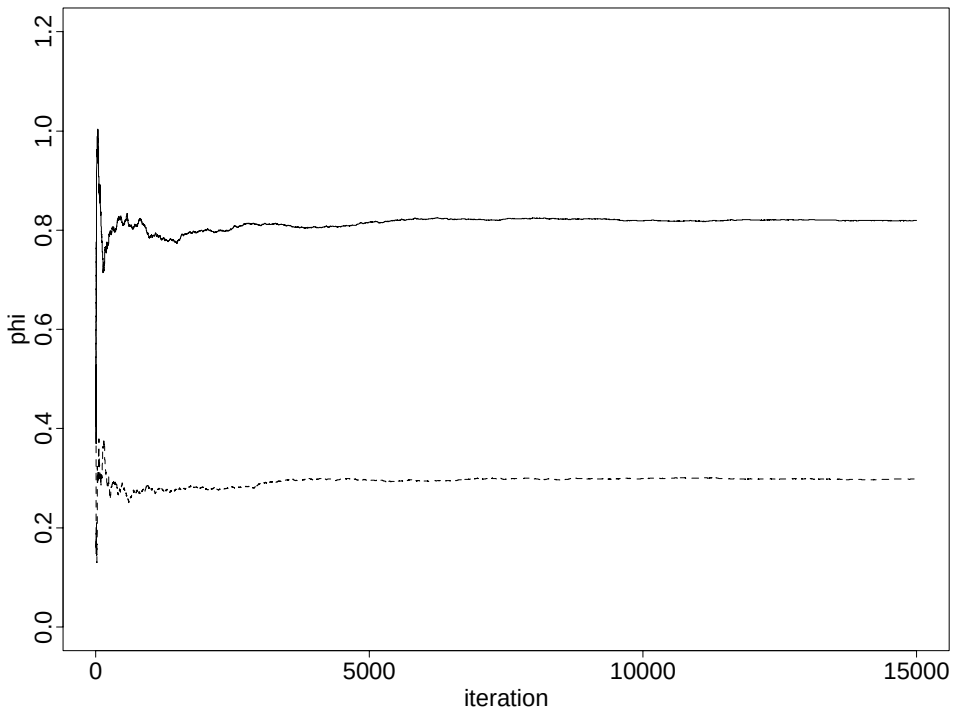
Se ve si la cadena está explorando bién el espacio o no.



Gráficos de las medias acumuladas (running means)

$$\widehat{\boldsymbol{\theta}}_t = \frac{1}{t}\sum_{j=1}^{t} \boldsymbol{\theta}^{(j)},$$

Cuando $J \to \infty$, $\widehat{\boldsymbol{\theta}}_t \to E[\boldsymbol{\theta}|\mathbf{x}]$ y entonces, mirando el gráfico, si se estabiliza, es una indicación de convergencia.



Diagnósticos formales de convergencia

Ver, por ejemplo Mengerson et al (1999).

Problemas de autocorrelación

Como se generan datos $\boldsymbol{\theta}^{(t)}$ de una cadena de Markov, la muestra es autocorrelada. Es conveniente hacer un gráfico de la función de autocorrelación estimada. Altas autocorrelaciones pueden significar convergencia lenta.

¿Qué hacer con los datos generados?

Se usa la primera parte (desde 5 a 50 %) de los datos generados por la cadena de Markov como datos de *burn in* para que la cadena olvide su estado inicial.

Si hay autocorrelación alta hasta un retardo r se puede considerar *thinning* los datos, es decir seleccionar sólo un dato en cada r para que la muestra sea incorrelada.

Se estima (por ejemplo) la media a posteriori de $\boldsymbol{\theta}$ mediante

$$E[\boldsymbol{\theta}|\mathbf{x}] \approx \frac{1}{N} \sum_{t=1}^{N} \boldsymbol{\theta}^{(t)}$$

donde N es el número de iteraciones (*thinned*) de la muestra después del periodo de *burn in*.

Varios algoritmos MCMC

Bajo distintas elecciones de las funciones de propuesta $g_i(\cdot|\boldsymbol{\theta})$, se puede simplificar el cálculo de las probabilidades de aceptación α_i. Existen varios algoritmos. A menudo, se utilizan distintos algoritmos para muestrear cada bloque $\boldsymbol{\theta}_i$.

El algoritmo Metropolis

Sea $g(\boldsymbol{\phi}|\boldsymbol{\theta}) = g(\boldsymbol{\theta}|\boldsymbol{\phi})$, una distribución simétrica. Entonces, se tiene

$$\alpha = \text{mín}\left[1, \frac{f(\boldsymbol{\phi})l(\boldsymbol{\phi}|\mathbf{x})}{f(\boldsymbol{\theta})l(\boldsymbol{\theta}|\mathbf{x})}\right]$$

Observación 81 *Es una versión de un algoritmo basado en un paseo aleatorio:*

$$\boldsymbol{\phi} = \boldsymbol{\theta}^{(t)} + \boldsymbol{\epsilon}^{(t)}.$$

Es decir que $g(\boldsymbol{\phi}|\boldsymbol{\theta}) = g(\boldsymbol{\phi} - \boldsymbol{\theta})$.

Ejemplo 117 *Sea* $X|\theta \sim \mathcal{C}(\theta, 1)$ *con una distribución a priori uniforme* $f(\theta) \propto 1$. *Dados los datos* $\mathbf{x}$, *se quiere simular una muestra de* $f(\theta|\mathbf{x})$.

Es posible usar una distribución de propuesta normal, por ejemplo, $g(\phi|\theta) = \mathcal{N}(\theta, s^2)$ *donde* s^2 *es la cuasi varianza de los datos.*

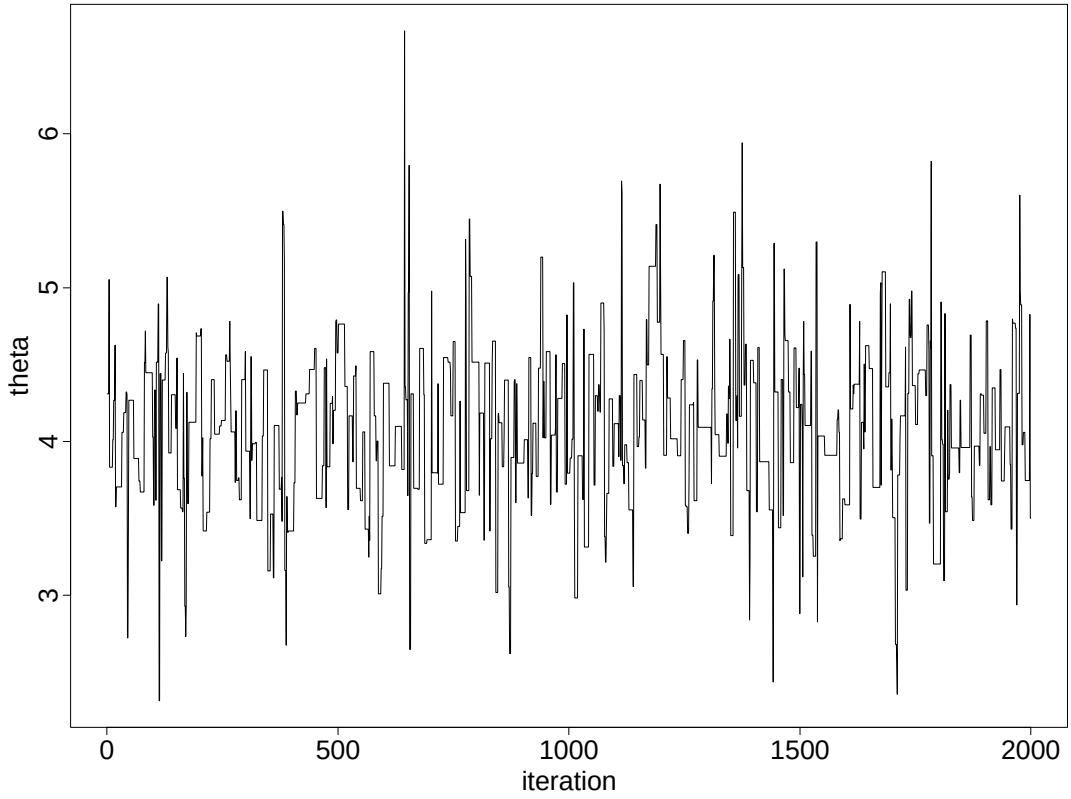
Entonces, $g(\phi|\theta) = g(\theta|\phi)$ *y la probabilidad de aceptar un candidato* ϕ *dado* $\theta^{(t)}$ *es*

$$\alpha = \prod_{i=1}^{n} \frac{1 + (x_i - \theta^{(t)})^2}{1 + (x_i - \phi)^2}.$$

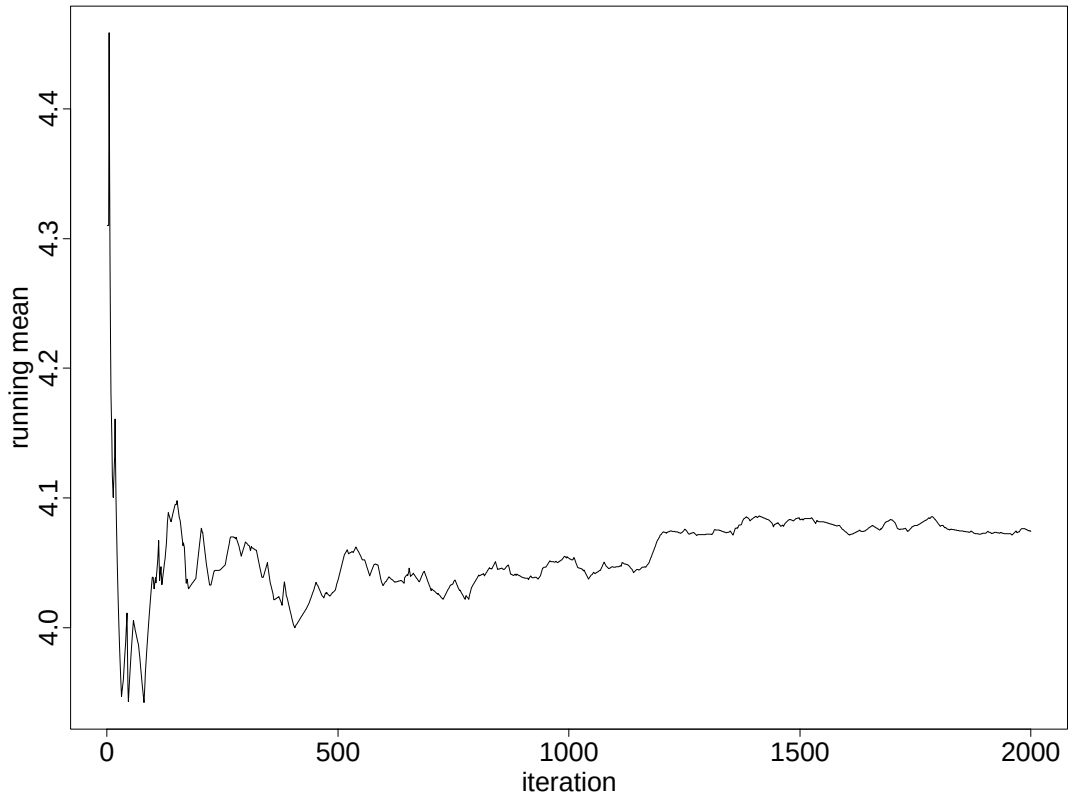
El diagrama ilustra los valores $\theta^{(t)}$ *generados en 2000 iteraciones del algoritmo Metropolis, dada la muestra*

$\mathbf{x} = (4{,}3, 2{,}2, 3{,}1, 8{,}4, -1{,}2, 6{,}7, 4{,}4, 7{,}3, 4{,}1, 3{,}8)$

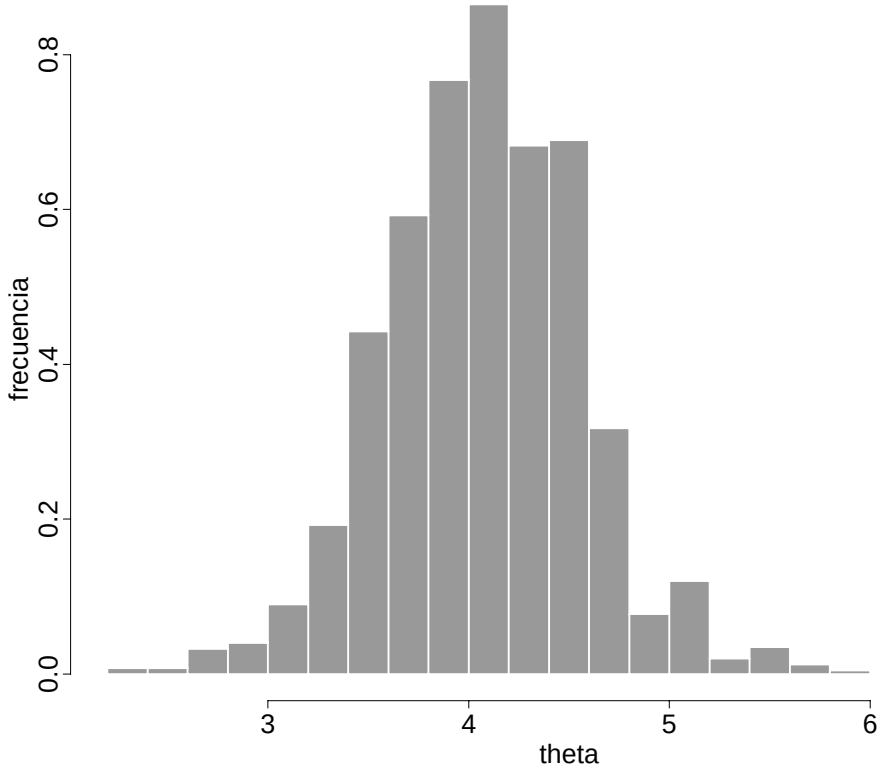
y el valor inicial $\theta^{(0)} = \bar{x}$.


theta
3
4
5
6
0
500
1000
1500
2000
iteration

El segundo diagrama muestra la media acumulada de θ frente el número de iteraciones. La media se ha estabilizado después de (aproximadamente) 1200 iteraciones.



El tercer diagrama muestra un histograma de los datos generados por el algoritmo.



El estimador de la media a posteriori es $E[\theta|\mathbf{x}] \approx$ 4,09 *y la tasa de aceptación es aproximadamente 23 %.*

El muestreo de independencia

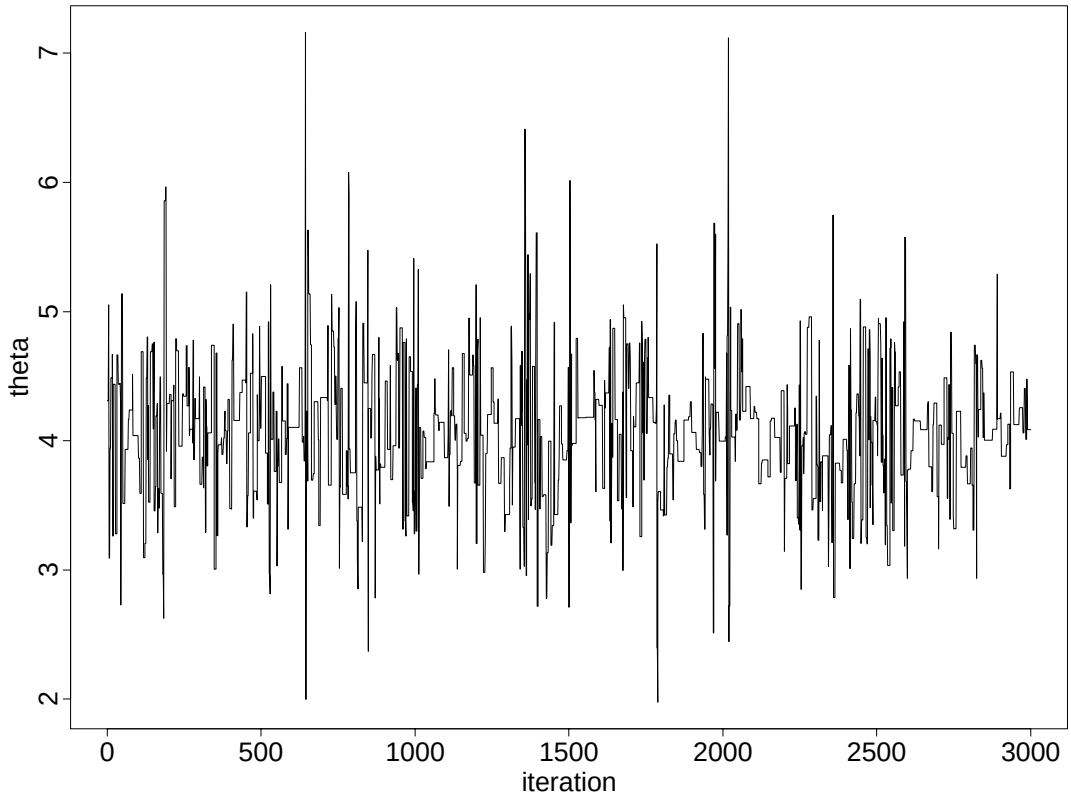
Otra posibilidad es suponer que $g(\boldsymbol{\phi}|\boldsymbol{\theta}) = g(\boldsymbol{\phi})$ independiente de $\boldsymbol{\theta}$.

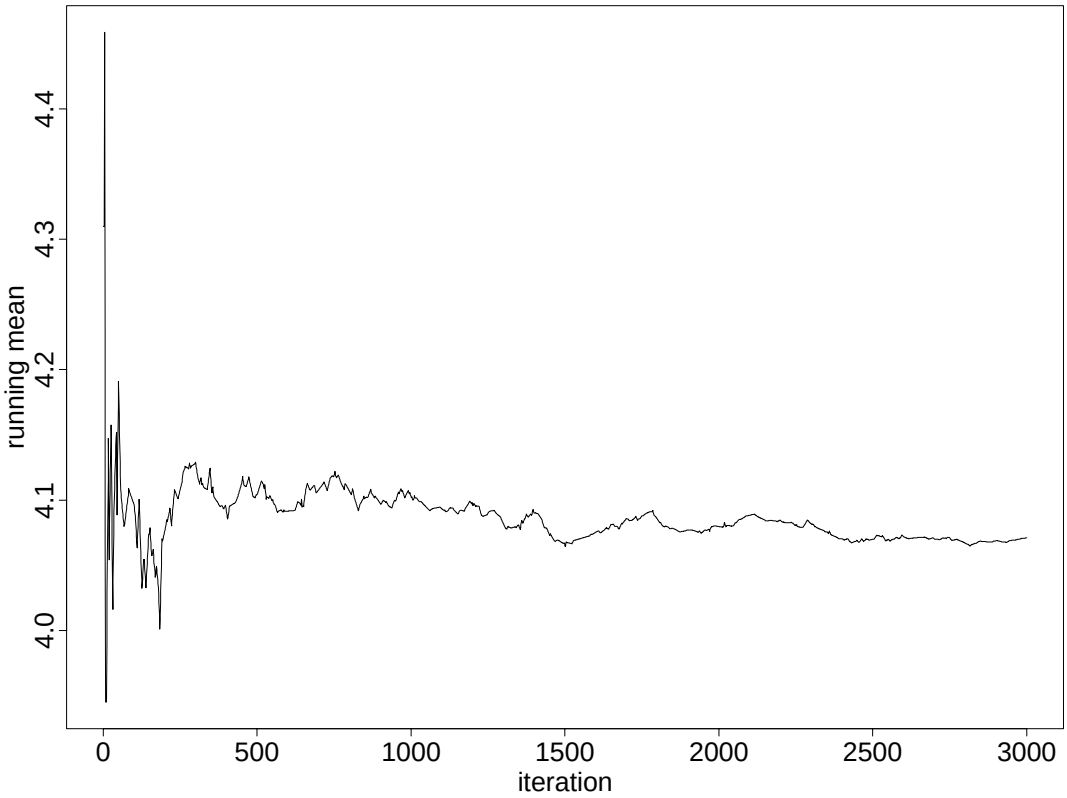
Ejemplo 118 *Volviendo a Ejemplo 117, sea ahora* $g(\phi) = \mathcal{N}(\bar{x}, s^2)$.

La probabilidad de aceptar ϕ *dado* $\theta^{(t)}$ *es*

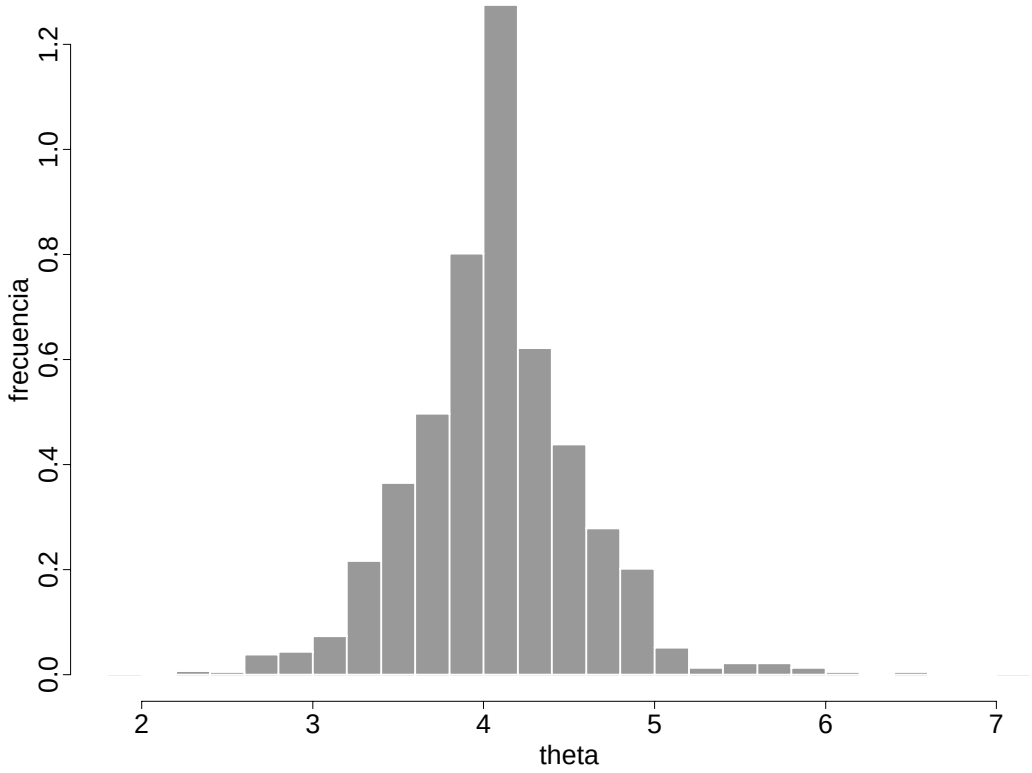
$$\alpha = e^{-\frac{1}{2s^2}[(\theta^{(t)}-\bar{x})^2-(\phi-\bar{x})^2]} \prod_{i=1}^{n} \frac{1+(x_i-\theta^{(t)})^2}{1+(x_i-\phi)^2}$$

Corriendo el algoritmo dados los datos anteriores, se tienen los siguientes resultados.


theta
2
3
4
5
6
7
0
500
1000
1500
2000
2500
3000
iteration



Se ve que la convergencia es menos rápida que para el método Metropolis.



El histograma es similar al anterior.

El estimador de la media a posteriori es $E[\theta|\mathbf{x}] \approx 4{,}06$ *con tasa de aceptación de 24 %.*

El muestreo Gibbs

El muestreo Gibbs (Geman y Geman 1984, Tanner y Wong 1987, Gelfand y Smith 1990) es una versión del método Metropolis Hastings con partición del espacio con

$$g_i(\boldsymbol{\phi}_i|\boldsymbol{\theta}_i, \boldsymbol{\theta}_{-i}) = f(\boldsymbol{\phi}_i|\boldsymbol{\theta}_{-i}, \mathbf{x})$$

la distribución condicional a posteriori.

Dada esta distribución, la probabilidad de aceptar el candidato $\boldsymbol{\phi}_i$ es

$$\begin{aligned} \alpha_i &= \text{mín}\left[1, \frac{f(\boldsymbol{\phi}_i|\boldsymbol{\theta}_{-i}^{(t)}) l(\boldsymbol{\phi}_i|\mathbf{x}, \boldsymbol{\theta}_{-i}^{(t)}) f\left(\boldsymbol{\theta}_i^{(t)}|\mathbf{x}, \boldsymbol{\theta}_{-i}^{(t)}\right)}{f(\boldsymbol{\theta}_i^{(t)}|\boldsymbol{\theta}_{-i}^{(t)}) l(\boldsymbol{\theta}_i^{(t)}|\mathbf{x}, \boldsymbol{\theta}_{-i}^{(t)}) f\left(\boldsymbol{\phi}_i|\mathbf{x}, \boldsymbol{\theta}_{-i}^{(t)}\right)}\right] \\ &= 1 \end{aligned}$$

es decir que los valores propuestos son siempre aceptados.

El algoritmo

El algoritmo para el muestreo Gibbs es:

1. Comenzar con valores iniciales arbitrarios $\boldsymbol{\theta}^{(0)}$

2. Generar $\boldsymbol{\theta}_1^{(t+1)} \sim f(\boldsymbol{\theta}_1|\mathbf{x}, \boldsymbol{\theta}_2^{(t)}, \ldots, \boldsymbol{\theta}_k^{(t)})$,

3. Generar $\boldsymbol{\theta}_2^{(t+1)} \sim f(\boldsymbol{\theta}_2|\mathbf{x}, \boldsymbol{\theta}_1^{(t+1)}, \boldsymbol{\theta}_3^{(t)}, \ldots, \boldsymbol{\theta}_k^{(t)})$,

4. $\vdots$

5. Generar $\boldsymbol{\theta}_k^{(t+1)} \sim f(\boldsymbol{\theta}_k|\mathbf{x}, \boldsymbol{\theta}_1^{(t+1)}, \ldots, \theta_{k-1}^{(t+1)})$,

6. Ir a 2.

Un ejemplo fácil

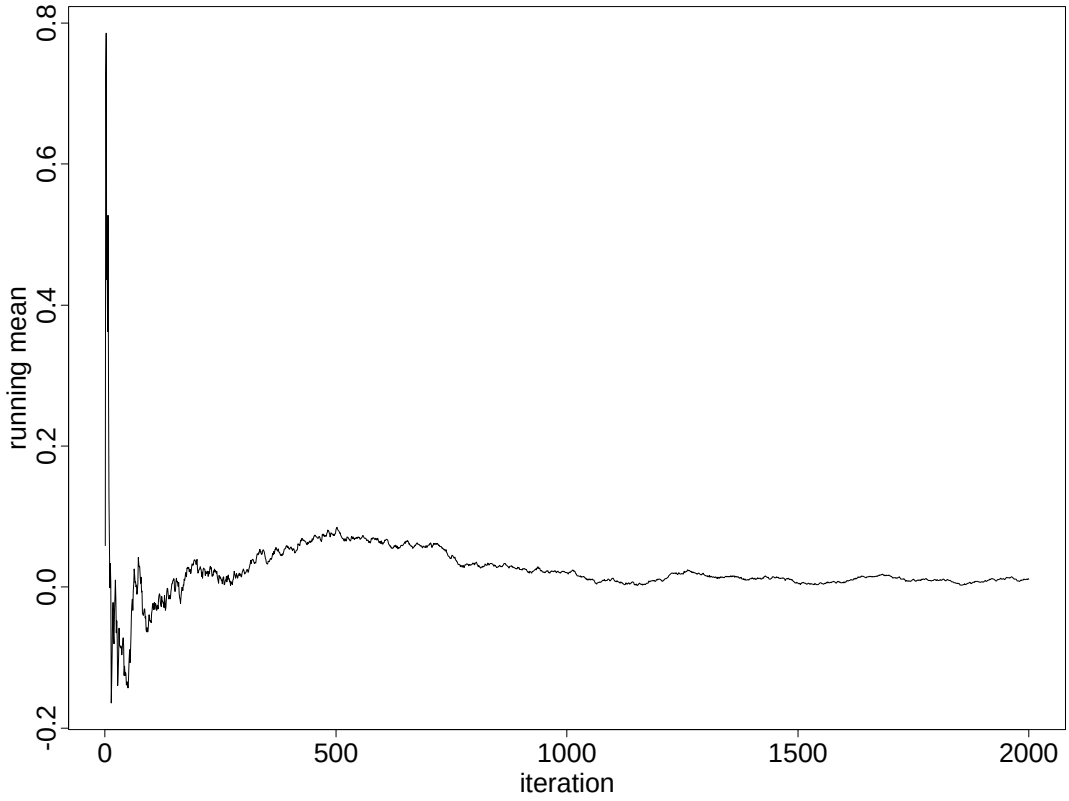
Ejemplo 119 *Volvamos al Ejemplo 112. También se puede utilizar una muestra de Gibbs para muestrear la distribución a posteriori.*

Las distribuciones condicionales en este caso son

$$\begin{aligned} \mu|\mathbf{x},\phi &\sim \mathcal{N}\left(\bar{x}, \frac{1}{n\phi}\right) \\ \phi|\mathbf{x},\mu &\sim \mathcal{G}\left(\frac{n-1}{2}, \frac{(n-1)s^2 + n(\mu-\bar{x})^2}{2}\right) \end{aligned}$$

y es muy fácil muestrear las dos.

El diagrama muestra el valor estimado de $E[\mu|\mathbf{x}]$ *frente al número de iteraciones. Se necesitan aproximadamente 1000 iteraciones para la convergencia.*



Observación 82 *Obviamente es menos eficaz utilizar el método de Gibbs cuando se puede utilizar un método Monte-Carlo simple.*

Ejemplo 120 Modelos con datos censurados

Supongamos que Z_i es la vida de una máquina i en un estudio. Pongamos una densidad $f(Z_i|\boldsymbol{\theta})$. Habitualmente, la duración del test es corta y sólo observamos unas pocas vidas enteras y las otras observaciones están truncadas en un tiempo T.

Entonces, los datos observados son

$$X_i = \begin{cases} Z_i & \text{si } X_i < T \\ T & \text{si } X_i > T \end{cases}$$

y la verosimilitud es

$$l(\boldsymbol{\theta}|\mathbf{x}) \propto \prod_{i=1}^{n_1} f(z_i|\boldsymbol{\theta})(1 - F(T|\boldsymbol{\theta}))^{n_2}$$

donde se supone que se observan n_1 tiempos de vida enteros y n_2 tiempos truncados a T.

A menudo la forma de la verosimilitud es complicada y no se pueden utilizar métodos sencillos para hacer la inferencia.

Pero se pueden simular las variables Z_i en un algoritmo Gibbs. Dado $X_i, \boldsymbol{\theta}$ tenemos $Z_i = X_i$ si $X_i < T$ y $f(Z_i|\boldsymbol{\theta}, X_i = T) \propto f(Z_i|\boldsymbol{\theta})$ truncado a $[T, \infty)$ si $X_i = T$. Es bastante fácil simular la distribución truncada usando por ejemplo el método de rechazo.

Observación 83 *Para simplificar el problema se han introducido los tiempos de vida no observados. Estas variables son variables latentes.*

Un algoritmo de Gibbs para evaluar la distribución a posteriori de $\boldsymbol{\theta}$ dados los datos $\mathbf{x}$ será el siguiente.

1. $t = 0$

2. Fijar valores iniciales $\boldsymbol{\theta}^{(0)}$.

3. Generar $z_{n_1+1}^{(t+1)}, \ldots, z_{n_1+n_2}^{(t+1)}$ de $f(z_i|x_i, \boldsymbol{\theta}^{(t)})$

4. Generar $\boldsymbol{\theta}^{(t+1)}$ de $f(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z}^{(t+1)}) = f(\boldsymbol{\theta}|\mathbf{z}^{(t+1)})$.

5. $t = t + 1$, Goto 3.

Ejemplo 121 *Podemos ilustrar el algoritmo del Ejemplo previo suponiendo que* $Z_i \sim \mathcal{E}(\theta)$. *En este caso, se puede evaluar la verosimilitud precisamente.*

Sea $\mathbf{x} = (x_1, \ldots, x_5, 1, 1)$ *con* 5 *observaciones no censuradas y* 2 *datos truncados en 1. Sea la suma de las observaciones no truncadas,* $\sum_{i=1}^{5} x_i = 3$.

Entonces, la verosimilitud será

$$\begin{aligned} l(\theta|\mathbf{x}) &\propto \theta^5 \exp\left(-\theta \sum_{i=1}^{5} x_i\right) \exp(-\theta)^2 \\ &\propto \theta^5 \exp(-5\theta) \end{aligned}$$

y dada una distribución a priori de Jeffreys, $f(\theta) \propto \frac{1}{\theta}$*, la distribución a posteriori es* $\theta|\mathbf{x} \sim \mathcal{G}(5,5)$*.*

Comparamos la distribución teórica con los resultados del algoritmo de Gibbs. Se tiene

$$\begin{aligned} f(z|\theta, z>1) &= \frac{\theta e^{-\theta z}}{e^{-\theta}} \\ &= \theta e^{-\theta(z-1)} \end{aligned}$$

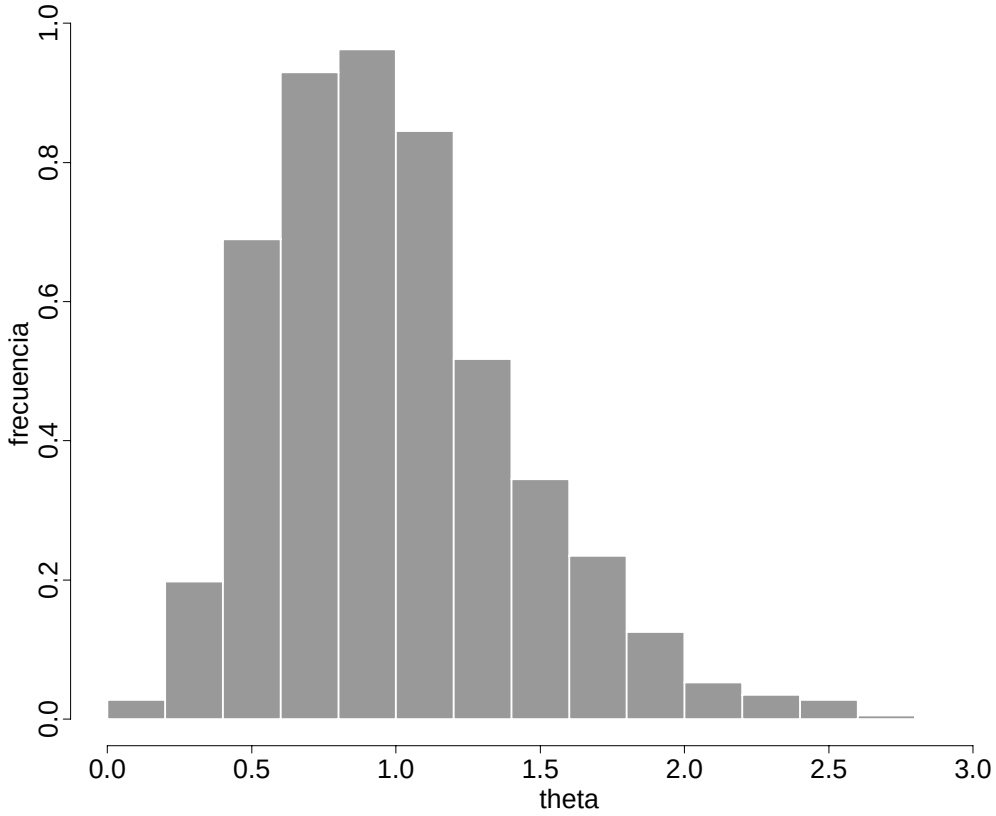
para $z > 1$ *que implica que*

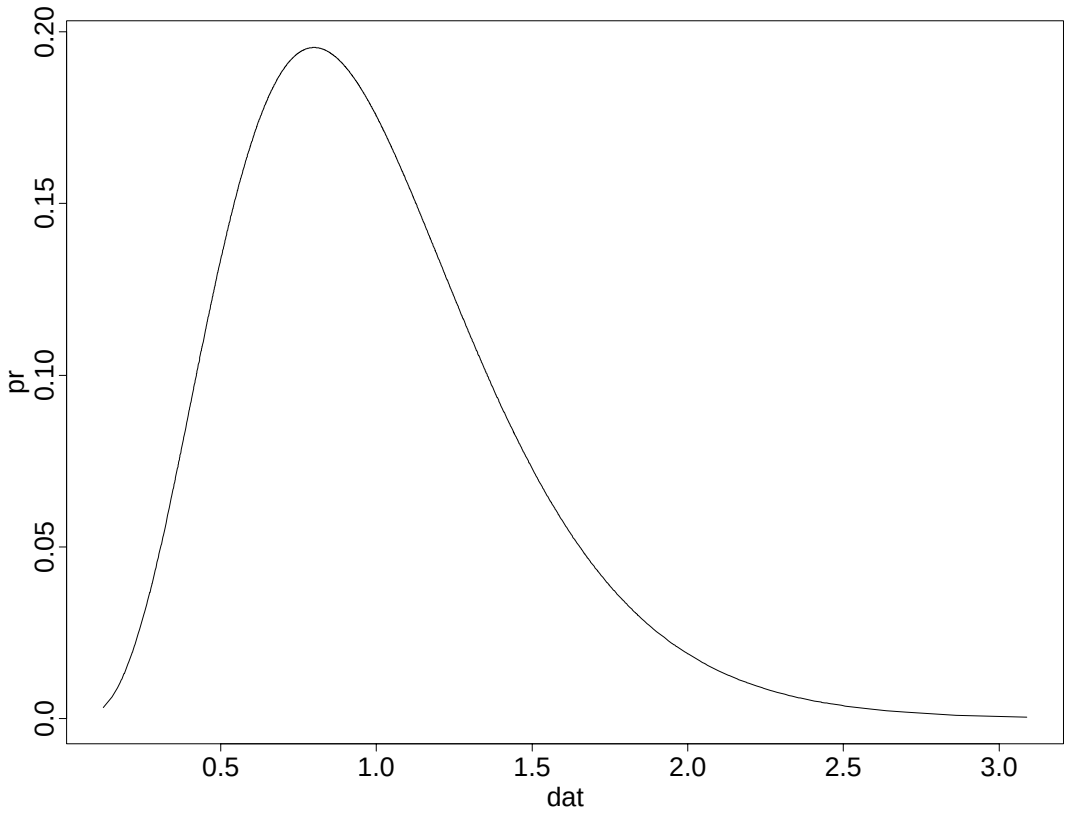
$$Z - 1|\theta, Z > 1 \sim \mathcal{E}(\theta)$$

y se puede muestrear esta distribución.

Además, $\theta|\mathbf{z} \sim \mathcal{G}(7, 3 + z_6 + z_7)$ *que es fácil de muestrear.*

Una muestra Gibbs con 1000 iteraciones en equilibrio generó los estimadores $E[\theta|\mathbf{x}] \approx 1{,}001$ *y* $V[\theta|\mathbf{x}] \approx {,}198$. *Los valores precisos son* $E[\theta|\mathbf{x}] = 1$ *y* $V[\theta|\mathbf{x}] = 0{,}2$. *El diagrama ilustra un histograma de los datos generados. Es parecido a la distribución* $\mathcal{G}(5, 5)$.




0.20
0.15
0.10
0.05
0.0
pr
0.5
1.0
1.5
2.0
2.5
3.0
dat

Ejemplo 122 Mixtura de normales

Supongamos la mixtura:

$$f(x|\boldsymbol{\theta}) = \sum_{i=1}^{k} w_i f_i(x|\mu_i, \sigma_i^2)$$

donde $\boldsymbol{\theta} = (w_1, \mu_1, \sigma_1^2, \ldots, w_k, \mu_k, \sigma_k^2)$ *y* $f_i(x|\mu_i, \sigma_i^2)$ *es una densidad normal con media* μ_i *y varianza* σ_i^2.

Sabemos que la verosimilitud es muy complicada y inferencia directa es imposible. Pero podemos simplificar el problema.

Dada X_j, *se define* Z_j:

$$P(Z_j = i|\mathbf{w}) = w_i \quad \textit{por}\, i = 1, \ldots, k$$

Entonces, tenemos la simplificación

$$f(x|\boldsymbol{\theta}, z) = f_z(x|\mu_z, \sigma_z^2)$$

Dados los datos $x_1, \ldots, x_n$*, la verosimilitud es*

$$\begin{aligned} l(\boldsymbol{\theta}, \mathbf{z}|\mathbf{x}) &\propto \prod_{j=1}^{n} \frac{1}{\sigma_{z_j}} \exp\left(-\frac{1}{2\sigma_{z_j}^2} \left(x_j - \mu_{z_j} \right)^2 \right) \\ &\propto \prod_{i=1}^{k} \frac{1}{\sigma_i^{n_i}} \exp\left(-\frac{1}{2\sigma_i^2} \sum_{j, z_j = i} (x_j - \mu_i)^2 \right) \end{aligned}$$

donde $n_i = \#\{z_j = i\}$ *y* $\sum_i n_i = n$*.*

Entonces, suponiendo distribuciones a priori

$$\begin{aligned} \mathbf{w} &\sim \mathcal{D}(\boldsymbol{\phi}) \quad \textit{Dirichlet} \\ \sigma_i^2 &\sim \mathcal{GI}(\alpha/2, \beta/2) \\ \mu_i|\sigma_i^2 &\sim \mathcal{N}(m_i, \sigma_i^2/c_i) \end{aligned}$$

podemos evaluar las distribuciones a posteriori condicionales:

Distribuciones a posteriori

$$
\begin{aligned}
P(Z_j = i|x_j, \boldsymbol{\theta}) &= \frac{w_i \frac{1}{\sigma_i} \exp\left(-\frac{1}{2\sigma_i^2}\left(x_j - \mu_i\right)^2\right)}{\sum_{i=1}^{k} w_i \frac{1}{\sigma_i} \exp\left(-\frac{1}{2\sigma_i^2}\left(x_j - \mu_i\right)^2\right)} \\
\mathbf{w}|\mathbf{x}, \mathbf{z} &\sim \mathcal{D}(\boldsymbol{\phi}^*) \\
\sigma_i^2|\mathbf{x}, \mathbf{z}, \mu_i &\sim \mathcal{GI}(\alpha_i^*/2, \beta_i^*/2) \\
\mu_i|\mathbf{x}, \mathbf{z}, \sigma_i^2 &\sim \mathcal{N}(m_i^*, \sigma_i^2/c_i^*)
\end{aligned}
$$

donde

$$
\begin{aligned}
\phi_i^* &= \phi_i + n_i \\
\alpha_i^* &= \alpha_i + n_i \\
\beta_i^* &= \beta_i + \sum_{j:z_j=i} (x_j - \mu)^2 \\
c_i^* &= c_i + n_i \\
m_i^* &= \frac{c m_i + n_i \bar{x}_i}{c_i + n_i}
\end{aligned}
$$

Entonces, se puede utilizar el muestreo de Gibbs para generar muestrear la distribución a posteriori $f(\boldsymbol{\theta}, \mathbf{z}|\mathbf{x})$.

Algoritmo

1. Fijar valores iniciales $\boldsymbol{\theta}^{(0)}$,

2. Generar Z_j de $P(Z_j|x_j, \boldsymbol{\theta}^{(i)})$ por $j = 1, \ldots, n$,

3. Calcular $\boldsymbol{\phi}^*$ y generar $\mathbf{w} \sim \mathcal{D}(\boldsymbol{\phi}^*)$,

4. Calcular m_i^*, c_i^* y generar $\mu_i \sim \mathcal{N}(m_i^*, \sigma_i^{2^{(i)}}/c_i^*)$,

5. Calcular α_i^*, β_i^* y generar $\sigma_i^2 \sim \mathcal{GI}(\alpha_i^*/2, \beta_i^*/2)$,

6. Ir a 2.

Algunas observaciones

- *No se pueden utilizar distribuciones a priori impropias porque se llevan a distribuciones a posteriori impropias. Está claro porque es posible que todos los $Z_j \neq i$ por algun elemento de la mixtura i.*

- *De vez en cuando, se reparametiza el problema porque la convergencia es bastante lenta. Ver Robert (1996).*

- *Existen métodos para comparar mixtures de tamaños distintos mediante factores Bayes. Ver por ejemplo Chib (1995).*

- *Se extiende el método al caso de k desconocido. Ver, por ejemplo Richardson y Green (1997) o Stephens (2000).*

Ejemplo 123 *Consideramos datos sobre la sensibilidad a luz de los ojos de 45 monos (Bowmaker et al 1985).*

Se supone que los datos provienen de una mixtura de dos normales con varianza común.

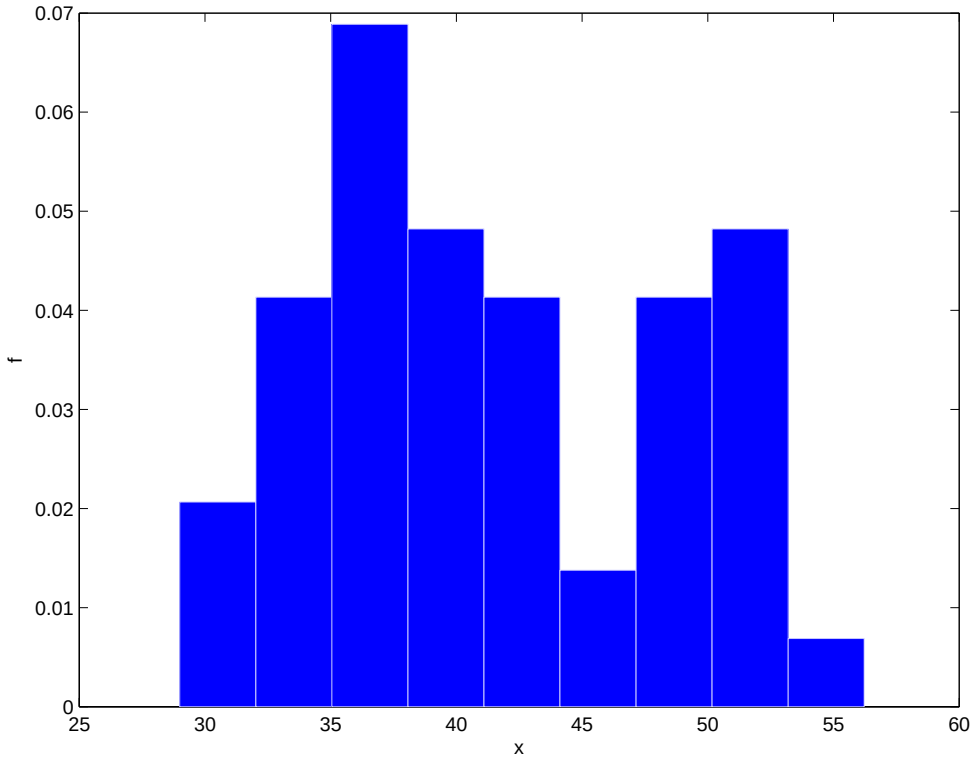
$$Y \sim p\mathcal{N}\left(\lambda_1, \sigma^2\right) + (1-p)\mathcal{N}\left(\lambda_2, \sigma^2\right)$$

donde $\lambda_2 > \lambda_1$.

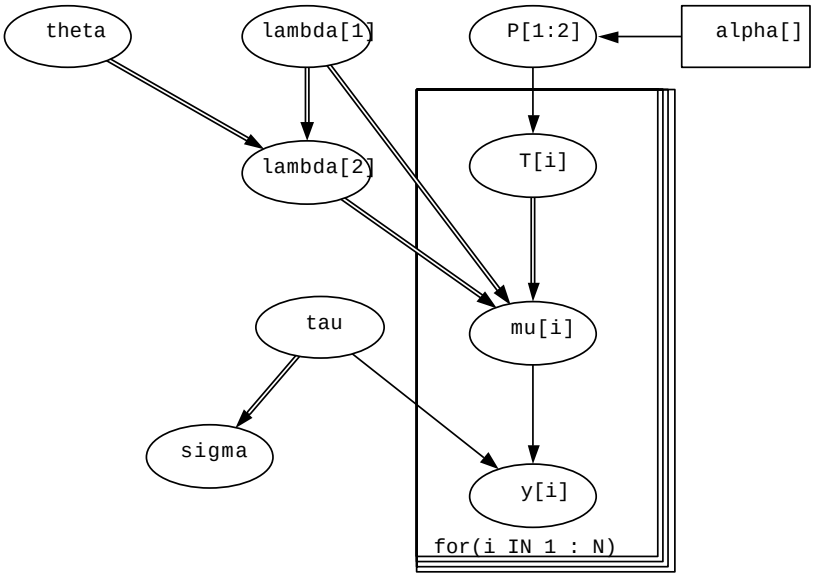
Para simplificar el cálculo en WinBugs, se define $\lambda_2 = \lambda_1 + \theta$ *donde* $\theta > 0$.

Se imponen distribuciones a priori poco informativas.

En primer lugar, se ve un histograma de los datos que ilustra la bimodalidad.



El segundo dibujo muestra la estructura del modelo.

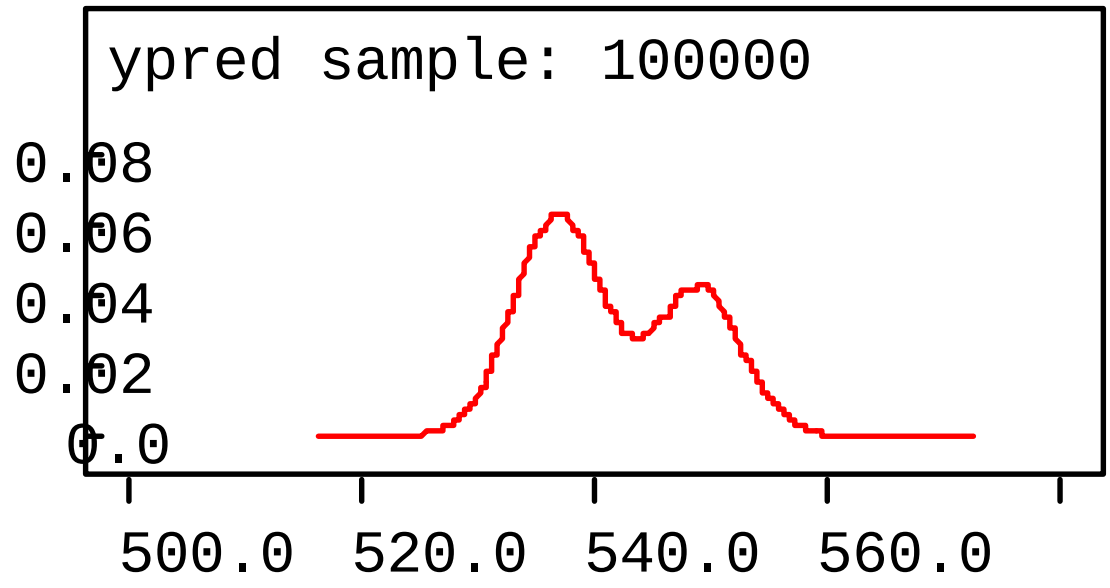

theta
lambda[1]
P[1:2]
alpha[]
lambda[2]
T[i]
tau
mu[i]
sigma
y[i]
for(i IN 1 : N)

Se utilizó una muestra de tamaño 100000 con 10000 iteraciones de burn in.

La tabla muestra las características de la distribución a posteriori.

node	mean	sd	MC error	2.5%	median	97.5%
P[1]	0.5992	0.08873	7.575E-4	0.4236	0.6022	0.7607
P[2]	0.4008	0.08873	7.575E-4	0.2393	0.3978	0.5764
lambda[1]	536.8	0.9679	0.007869	535.0	536.7	538.7
lambda[2]	548.9	1.317	0.01275	546.2	548.9	551.2
sigma	3.789	0.65	0.0076	2.927	3.672	5.465
Ypred	541.6	7.11	0.01999	530.0	540.3	554.9

En la siguiente transparencia se ve un estimador kernel de la densidad predictiva de Y.


ypred sample: 100000
0.08
0.06
0.04
0.02
0.0
500.0
520.0
540.0
560.0

Otros algoritmos

- ARMS. Método Metropolis Hastings con rechazo adaptivo. Ver Gilks et al (1995).

  Es un algoritmo basado en la combinación del algoritmo de rechazo adaptivo con métodos Metropolis Hastings. Este algoritmo es la base para el muestreo Gibbs en *WinBugs*.

- Slice sampler. (Neal 2000).

  Se quiere muestrear $f(\boldsymbol{\theta}|\mathbf{x}) \propto g(\boldsymbol{\theta})$. Se emplea un algoritmo basado en aumentación los datos.

El algoritmo es

1. $t = 0$. `Valor inicial` $\boldsymbol{\theta}^{(0)}$.
2. `Muestrear` $u^{(t+1)} \sim U[0, g(\boldsymbol{\theta}^{(t)})]$
3. `Muestrear` $\boldsymbol{\theta}^{(t+1)} \sim U\left(A^{(t+1)}\right)$ `donde` $A^{(t+1)} = \left\{\boldsymbol{\theta};\ g(\boldsymbol{\theta}) \geq u^{(t+1)}\right\}$.
4. $t = t + 1$. `Ir a 2.`

Más o menos fácil de programar.

- Difusiones Langevin. (Grenander y Miller 1994). MCMC híbrido (Neal 1996).

  Otros algorítmos basados en aumentación de variables.

- Salto reversible. (Green 1995).

  Extiende el algoritmo Metropolis Hastings a problemas de dimensión variable.

- Muestreo perfecto.

  Un problema de los algoritmos de MCMC es que es difícil decidir cuando ha convergido la cadena. Pero en algunos artículos recientes, se ha introducido el muestreo perfecto: una versión de MCMC para que se pueda estar seguro que se tiene una muestra de la distribución de equilibrio de la cadena.

  El algoritmo tipo usa la idea de *coupling from the past* para determinar la llegada en equilibrio. Ver Casella et al (2000) o `dimacs.rutgers.edu/~dbwilson/exact.html/`.

- Y muchos más. Ver, por ejemplo, Andreu et al (2003).