

Tarea 8. MCMC: MH con Kerneles Híbridos y Gibbs Sampler

Ricardo Chávez Cáliz

November 1, 2017

Problema 1. Simule valores usando el algoritmo de Metropolis-Hastings para $f(x, y)$ la distribución normal bivariada con media μ y matriz de varianza Σ usando Kerneles híbridos.

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

Use las siguientes propuestas:

$$q_1 \left(\begin{pmatrix} x_p \\ y_p \end{pmatrix} \middle| \begin{pmatrix} x \\ y \end{pmatrix} \right) = f_{X|Y}(x_p|y) \cdot \mathbb{1}_{(y_p=y)}$$

$$q_2 \left(\begin{pmatrix} x_p \\ y_p \end{pmatrix} \middle| \begin{pmatrix} x \\ y \end{pmatrix} \right) = f_{Y|X}(y_p|x) \cdot \mathbb{1}_{(x_p=x)}$$

Considere los casos $\rho = 0.8$ y $\rho = 0.99$.

Se aplicó el algoritmo solicitado para la función especificada el cual puede ser encontrado en la función `NormalMHMC` del código adjunto. Se fijaron como puntos iniciales los correspondientes a la media, ie $x_0 = \mu_1$ y $y_0 = \mu_2$.

Para verificar la simulación se graficaron las curvas de nivel de $f(x, y)$ y se sobrepusieron los valores de la muestra obtenida mediante el algoritmo MH con los kernels híbridos propuestos.

Sin perdida de generalidad se considera el vector 0 como μ . Eligiendo cada propuesta de manera uniforme, es decir $\omega_1 = \frac{1}{2}$, se obtuvieron los siguientes resultados con distintos tamaños de muestra $n = 1000$ y $n = 10000$.

Para $\rho = 0.8$, se puede observar en la figura 1 que a medida que se aumenta el número de la muestra la aproximación a $f(x, y)$ es mejor.

Para $\rho = 0.99$ (ver figura 2), a diferencia del caso anterior, la muestra no se aproxima a $f(x, y)$ a pesar de tener una gran cantidad de elementos. Esto tiene sentido dado que las propuestas que se están tomando son las marginales en dirección horizontal y vertical, por lo que la manera en que se mueve la cadena a cada paso depende de ρ y a medida que ρ tiende a 1, los pasos son más pequeños, sobre todo acercándose a las colas de $f(x, y)$ (ver las curvas de nivel de $f(x, y)$). Por lo tanto empezando en la media, será necesario dar una gran cantidad de pasos (una muestra muy grande) para poder obtener información de las colas de la densidad.

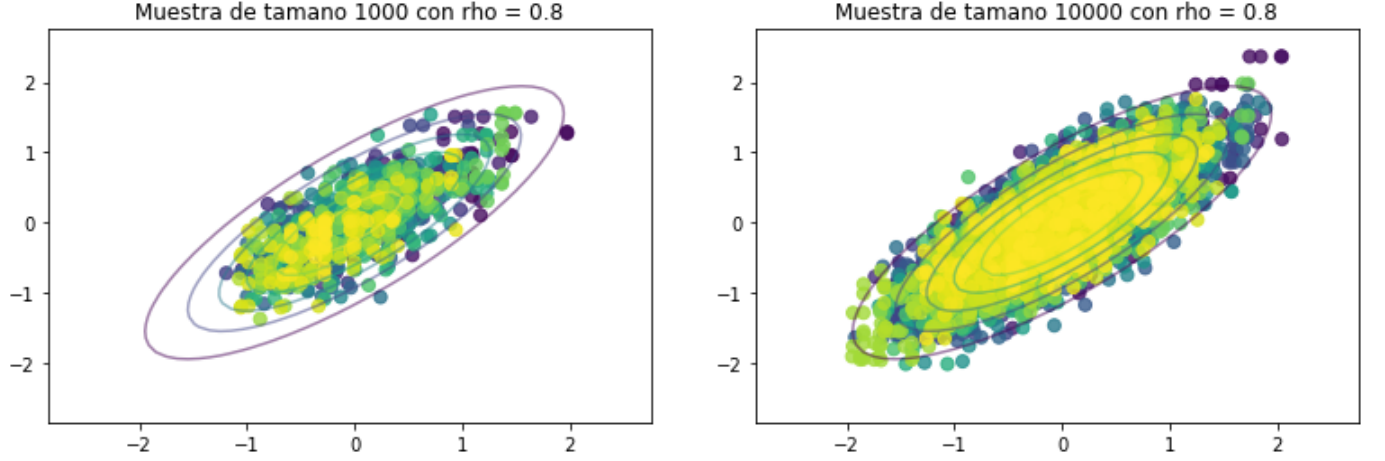


Figure 1: Se muestra las curvas de nivel de $f(x, y)$ y una muestra de tamaño 1000 con $\rho = 0.8$

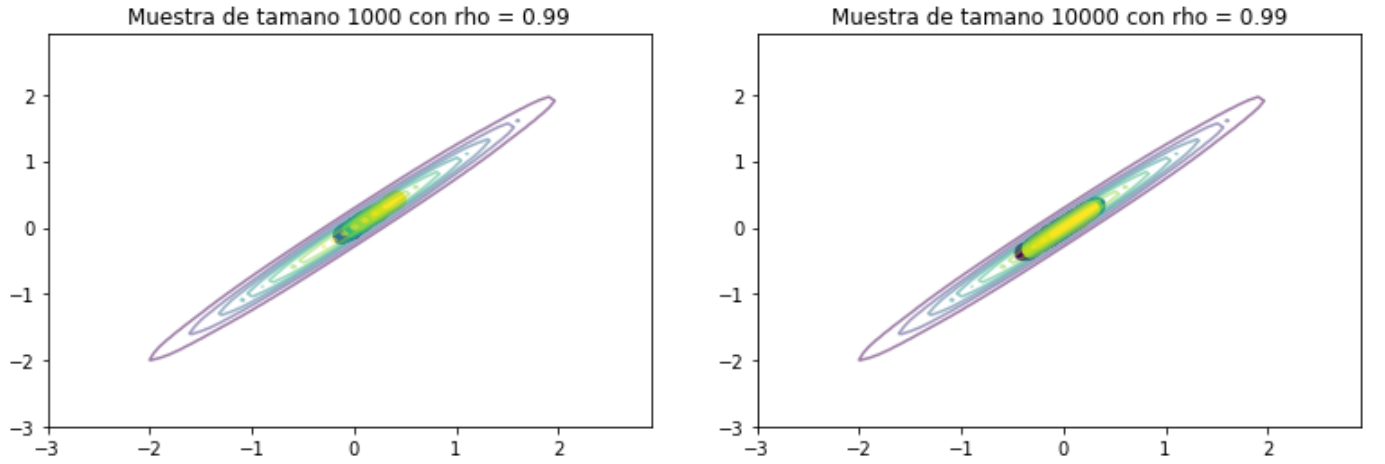


Figure 2: Se muestra las curvas de nivel de $f(x, y)$ y una muestra de tamaño 1000 con $\rho = 0.99$

Problema 2. Considere t_1, \dots, t_n tiempos de falla, con distribución $Weibull(\alpha, \lambda)$, es decir $f(t_i|\alpha, \lambda) = \alpha \lambda t_i^{\alpha-1} e^{-t_i^\alpha \lambda}$ donde α y λ se asumen con densidades apriori: $\alpha \sim \exp(c)$ y $\lambda|\alpha \sim Gama(\alpha, b)$. Así se obtiene la distribución posterior

$$f(\alpha, \lambda|\bar{t}) \propto f(\bar{t}|\alpha, \lambda)f(\alpha, \lambda)$$

A partir del algoritmo MH usando Kernels híbridos simule valores de la distribución posterior $f(\alpha, \lambda|\bar{t})$, considerando las siguientes propuestas:

Propuesta 1: $\lambda_p|\alpha, \bar{t} \sim Gama\left(\alpha + n, b + \sum_{i=1}^n t_i^\alpha\right)$ y dejando α fijo.

Propuesta 2: $\alpha_p|\lambda, \bar{t} \sim Gama(n + 1, -\log(b) - \log(r_1) + c)$, con $r_1 = \prod_{i=1}^n t_i$ y dejando λ fijo.

Propuesta 3: $\alpha_p \sim \exp(c)$ y $\lambda_p | \alpha_p \sim \text{Gama}(\alpha_p, b)$.

Propuesta 4 (RWMH): $\alpha_p = \alpha + \epsilon$, con $\epsilon \sim N(0, \sigma)$ y dejando λ fijo.

Simular datos usando $\alpha = 1$ y $\lambda = 1$ con $n = 20$. Para la a priori usar $c = 1$ y $b = 1$.

Se aplicó el algoritmo solicitado para la función especificada el cual puede ser encontrado en la función `TiemposMHMC` del código adjunto.

Para verificar la simulación se graficaron las curvas de nivel de $\ln(f(\alpha, \lambda | \bar{t}))$ y se sobrepusieron los valores de la muestra obtenida mediante el algoritmo MH con los kernels híbridos propuestos. Ver figura 3.

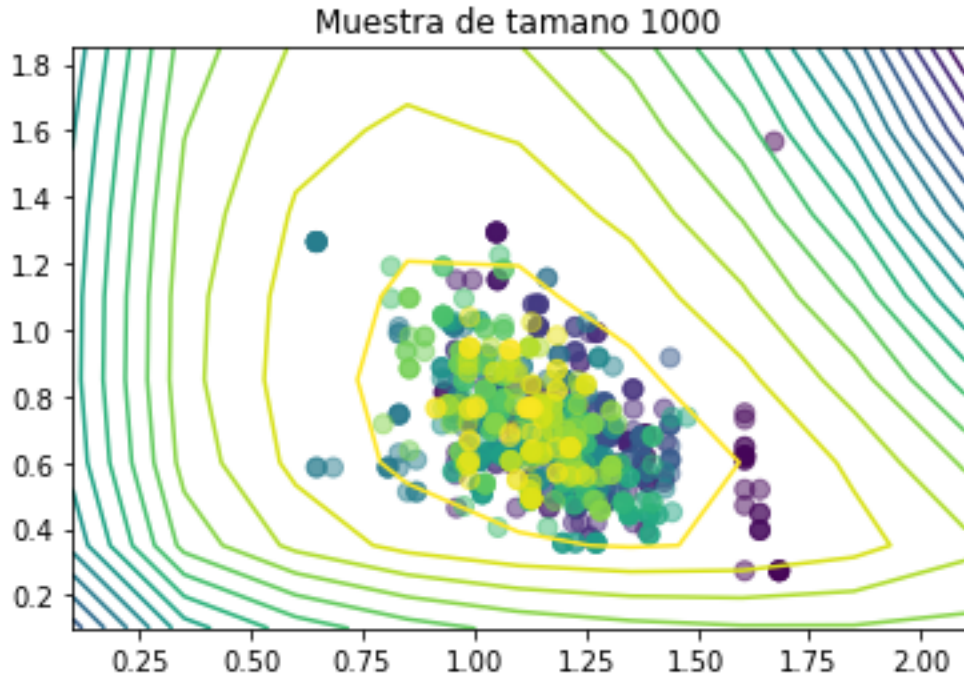


Figure 3: Se muestra las curvas de nivel de $\ln(f(\alpha, \lambda | \bar{t}))$ y una muestra de tamaño 1000

Para obtener los valores numéricos apropiados para ρ_2, ρ_3, ρ_4 se hicieron consideraciones de cálculos logarítmicos, para el primer kernel tenemos $\rho_1 = 1$, por lo tanto para esta propuesta no fue necesario. Se obtuvieron:

$$\rho_2 = e^{\min\{0, \ln(\Gamma(\alpha)) + \lambda(\alpha_p - \alpha) - \lambda \cdot (\sum_{i=1}^n t_i^{\alpha_p}) - \ln(\Gamma(\alpha_p)) + \lambda \cdot (\sum_{i=1}^n t_i^{\alpha})\}}$$

$$\rho_3 = e^{\min\{0, n \cdot \ln(\frac{\alpha_p \cdot \lambda_p}{\alpha \cdot l}) + (\alpha_p - \alpha) \cdot \ln(r_1) - \lambda_p \cdot (\sum_{i=1}^n t_i^{\alpha_p}) + \lambda \cdot (\sum_{i=1}^n t_i^{\alpha})\}}$$

$$\rho_4 = e^{\min\{0, w\}}$$

donde $w = \alpha_p \cdot \ln(\lambda) - \lambda \cdot (\beta + \sum_{i=1}^n t_i^{\alpha_p}) + \alpha_p \cdot \ln(\beta) + n \cdot \ln(\alpha_p) + (\alpha_p - 1) \cdot (\sum_{i=1}^n \ln(t_i)) - \gamma \cdot \alpha_p - \ln(\Gamma(\alpha_p)) - \alpha \cdot \ln(\lambda) + \lambda \cdot (\beta + \sum_{i=1}^n t_i^{\alpha}) - \alpha \cdot \ln(\beta) - n \cdot \ln(\alpha) - (\alpha - 1) \cdot (\sum_{i=1}^n \ln(t_i)) + \gamma \cdot \alpha + \ln(\Gamma(\alpha))$

El algoritmo lleva registro de la cantidad de propuestas rechazadas, en la simulación aquí mostrada se rechazaron el 51.56% de las propuestas

Dependiendo de la muestra a veces es necesario calcular el logaritmo de números cercanos a cero para las curvas de nivel, por lo que se obtiene en algunas ocasiones `RuntimeWarning: divide by zero encountered in log` y las curvas de nivel pueden llegar a mostrarse extrañas, pero siempre simulando de manera correcta.

Problema 3. Considere el ejemplo referente al número de fallas de bombas de agua en una central nuclear donde p_i representa el número de fallas en el tiempo de operación t_i , con $i = 1, \dots, n$.

Se considera el modelo $p_i \sim \text{Poisson}(\lambda_i t_i)$, (las λ_i son independientes entre si), con distribuciones a priori $\lambda_i | \beta \sim \text{Gama}(\alpha, \beta)$ y $\beta \sim \text{Gama}(\gamma, \delta)$, por lo tanto:

$$f(\lambda_1, \dots, \lambda_n, \beta) = f(\lambda_1 | \beta) f(\lambda_2 | \beta) \dots f(\lambda_n | \beta) f(\beta)$$

Para la distribución posterior se tiene:

$$f(\lambda_1, \dots, \lambda_n, \beta | \bar{p}) \propto L(\bar{p}, \bar{\lambda}, \beta) f(\lambda_1, \dots, \lambda_n, \beta)$$

Simule valores de la distribución posterior $f(\lambda_1, \dots, \lambda_n, \beta | \bar{p})$, usando un kernel híbrido, considerando las propuestas:

$$\lambda_i | \bar{\lambda}_{-i}, \beta, \bar{t} \sim \text{Gama}(t_i p_i + \alpha, \beta + 1)$$

$$\beta | \bar{\lambda}, \bar{t} \sim \text{Gama}\left(n\alpha + \gamma, \delta + \sum_{i=1}^n \lambda_i\right).$$

Verifique que estas son propuestas Gibbs.

Use los datos del Cuadro 1 con los parámetros a priori $\alpha = 1.8$, $\gamma = 0.01$ y $\delta = 1$.

Bomba (i)	1	2	3	4	5	6	7	8	9	10
T. de uso (t_i)	94.32	15.72	62.88	125.76	5.24	31.44	1.05	1.05	2.1	10.48
# de fallas (p_i)	5	1	5	14	3	19	1	1	4	22

Table 1: Datos de bombas de agua en centrales nucleares (Robert y Casella, p. 385) para el ejemplo 8.3.

Se aplicó el algoritmo solicitado para la función especificada, pero se cambiaron las propuestas planteadas, debido a que no aproximaban lo debido. Más adelante se hace una justificación de dicho cambio. El algoritmo puede ser encontrado en la función `bombasAguaMHMC` del código adjunto.

El modelado está basado en la suposición de que los fallos de cada bomba siguen una distribución de Poisson cuyo parámetro es proporcional al tiempo de uso de cada bomba.¹.

¹Robert y Casella ejemplo 8.3)

Dichas constantes de proporcionalidad λ_i son las que pretendemos estimar con este método. Con las distribuciones a priori dadas, tenemos la distribución conjunta:

$$\begin{aligned} & \pi(\lambda_1, \dots, \lambda_{10}, \beta | t_1, \dots, t_{10}, p_1, \dots, p_{10}) \\ & \propto \prod_{i=1}^{10} \{ (\lambda_i t_i)^{p_i} \cdot e^{-\lambda_i t_i} \cdot \lambda_i^{\alpha-1} \cdot e^{-\beta \lambda_i} \} \beta^{10\alpha} \beta^{\gamma-1} e^{-\delta \beta} \\ & \propto \prod_{i=1}^{10} \{ \lambda_i^{p_i+\alpha-1} \cdot e^{-(t_i+\beta)\lambda_i} \} \beta^{10\alpha+\gamma-1} e^{-\delta \beta} \end{aligned}$$

de esta manera una descomposición natural para π en distribuciones condicionales es

$$\lambda_i | \bar{\lambda}_{-i}, \beta, \bar{t} \sim Gama(p_i + \alpha, \beta + t_i)$$

$$\beta | \bar{\lambda}, \bar{t} \sim Gama\left(n\alpha + \gamma, \delta + \sum_{i=1}^n \lambda_i\right).$$

Por la tanto se usaron estas propuestas para la simulación y los resultados se muestran en las figuras 4 y 5. En las que se grafica la simulación de cada parametro a estimar y se toma el promedio, el cual tambien se grafica y se puede apreciar como la muestra para cada λ y para β se acumula alrededor de promedio.

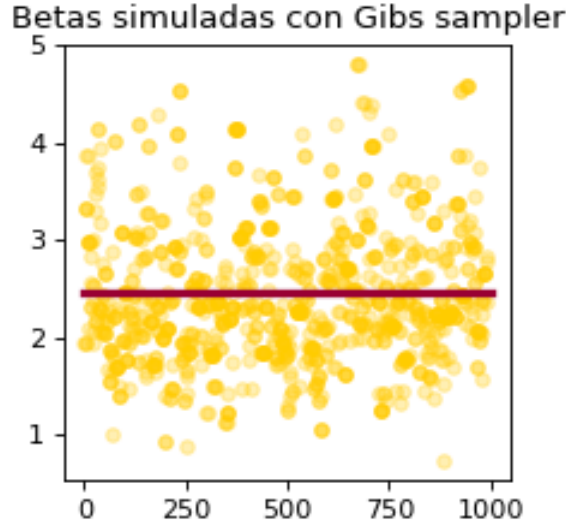


Figure 4: Se grafican las β obtenidas en una muestra de tamaño 1000 y se grafica la línea correspondiente al promedio muestral

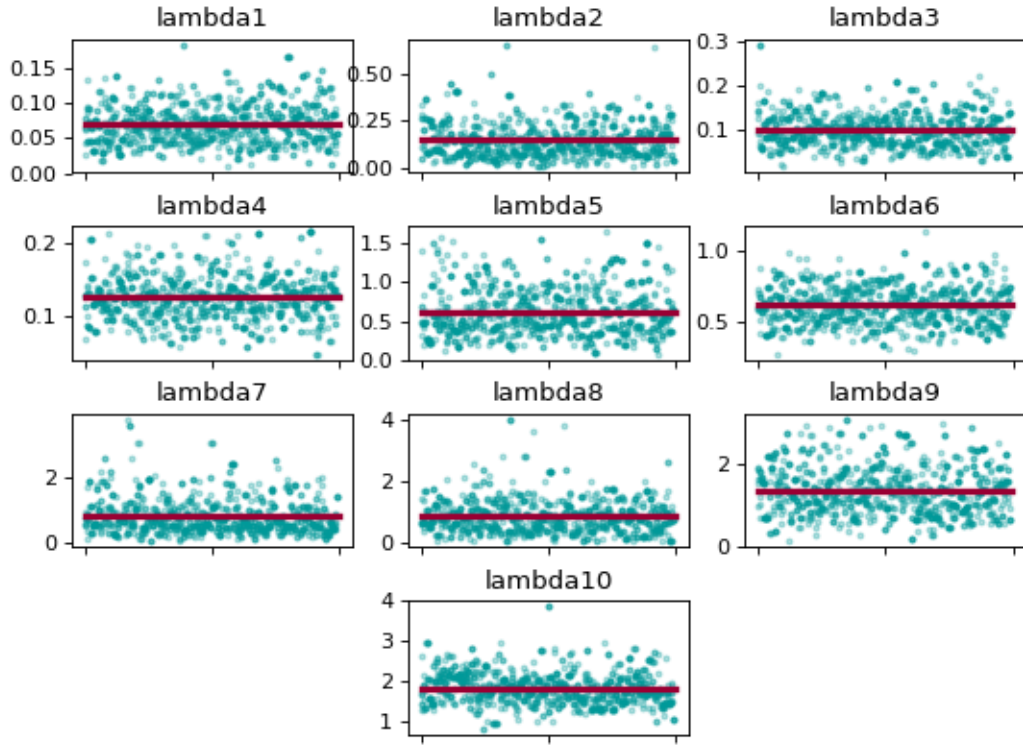


Figure 5: Se grafica cada λ_i para una muestra de tamaño 1000, y se grafica la línea correspondiente al promedio muestral para cada λ_i

Parámetro	media muestral
λ_1	0.0705370857746
λ_2	0.149854183199
λ_3	0.102397970937
λ_4	0.125647388644
λ_5	0.622845921325
λ_6	0.613016512921
λ_7	0.803668677919
λ_8	0.857987771718
λ_9	1.37131964859
λ_{10}	1.83555326618
β	2.45226496655

Table 2: Tabla de promedios