

# Cómputo Científico

## Tarea VIII

Joel Chacón Castillo  
*Guanajuato, México*

---

### 1. Simulación Normal bivariada

Aplique el algoritmo de Metropolis-Hastings considerando como función objetivo la distribución normal bivariada:

$$f_{x_1, x_2}(x) = \frac{1}{2\pi} |\Sigma|^{\frac{-1}{2}} \exp\left\{-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right\} \quad (1)$$

donde

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \quad \Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \quad (2)$$

Donde se tienen las siguientes distribución condicionales:

$$\begin{aligned} X_1|X_2 = x_2 &\sim N\left(\mu_1 + \rho\frac{\sigma_1}{\sigma_2}(x_2 - \mu_2), \sigma_1^2(1 - \rho^2)\right) \\ X_2|X_1 = x_1 &\sim N\left(\mu_2 + \rho\frac{\sigma_2}{\sigma_1}(x_1 - \mu_1), \sigma_2^2(1 - \rho^2)\right) \end{aligned} \quad (3)$$

considere las siguientes propuestas:

$$\begin{aligned} q_1((x_1', x_2')|(x_1, x_2)) &= f_{X_1|X_2}(x_1'|x_2)\mathbb{1}(x_2' = x_2) \\ q_2((x_1', x_2')|(x_2, x_1)) &= f_{X_2|X_1}(x_2'|x_1)\mathbb{1}(x_1' = x_1) \end{aligned} \quad (4)$$

A partir del algoritmo MH usando Kernels híbridos simule valores de la distribución normal bivariada, fijando  $\sigma_1 = \sigma_2 = 1$ , considere los casos donde  $\rho = 0,8$  y  $\rho = 0,99$ .

### Comentarios

Primero se establece una relación entre el método de Metrópolis Hastings y el método de muestreo de Gibbs mencionado en la pag. 381 de [1] (esto también se vió en clase).

**Theorem 1.** EL método de muestreo de Gibbs es equivalente a la composición de P-algoritmos de Metrópolis Hastings, con probabilidad de aceptación uniformemente igual a la unidad.

En consecuencia nunca se rechaza el criterio de aceptación para este caso,

La implementación está basada en el *Random Scan Gibbs Sampler* donde cada kernel se selecciona con probabilidad  $\omega$ , como se indica en la siguiente ecuación:

$$\begin{aligned} K &= \sum_{i=0}^n \omega_i k_i \\ s.a. \quad &\sum_i \omega_i = 1 \end{aligned} \tag{5}$$

Se aclara que en el código no se aplica el kernel cero para forzar que la cadena sea fuertemente aperiódica.

En la figura 2 ( $\rho = 0,8$ ) se ilustra el desplazamiento de la cadena de markov en cada una de las dos variables, y en la parte inferior el mapa de contorno con el recorrido realizado por la cadena. Por otra parte en la figura 1 se observa la simulación con una configuración distinta  $\rho = 0,99$ . Principalmente se puede observar que al inducir un grado de dependencia elevado entre las variables causa dificultades de que se pueda simular adecuadamente una distribución, principalmente esto sucede ya que el recorrido que realiza la cadena de markov es únicamente en orientación a los ejes (en este caso), no obstante se podría indicar una dirección mas favorecedora (en la esfera unitaria por ejemplo) que permita que la cadena de markov visite estados en el soporte completo, es decir, una convergencia mas acelerada. Una dirección podría estar relacionada con los vectores propios, como se sugiere en la tesis *Distribución de direcciones en el Gibbs sampler generalizado* por José Cricelio Montesinos.

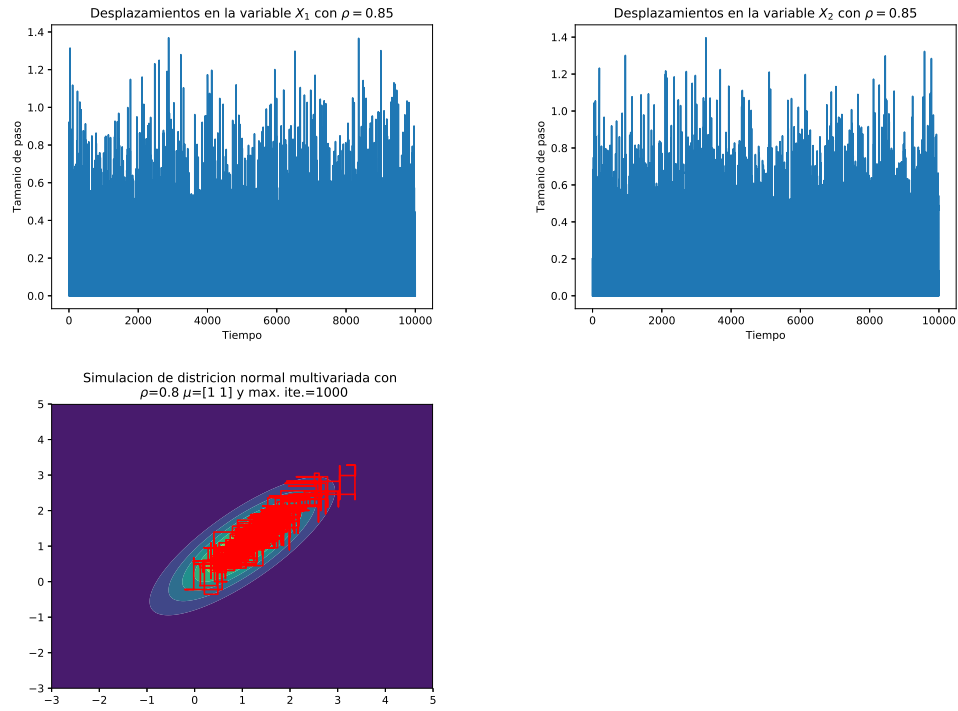


Figura 1: Desplazamientos realizados por la cadena de markov en la parte superior y el recorrido en la parte inferior.

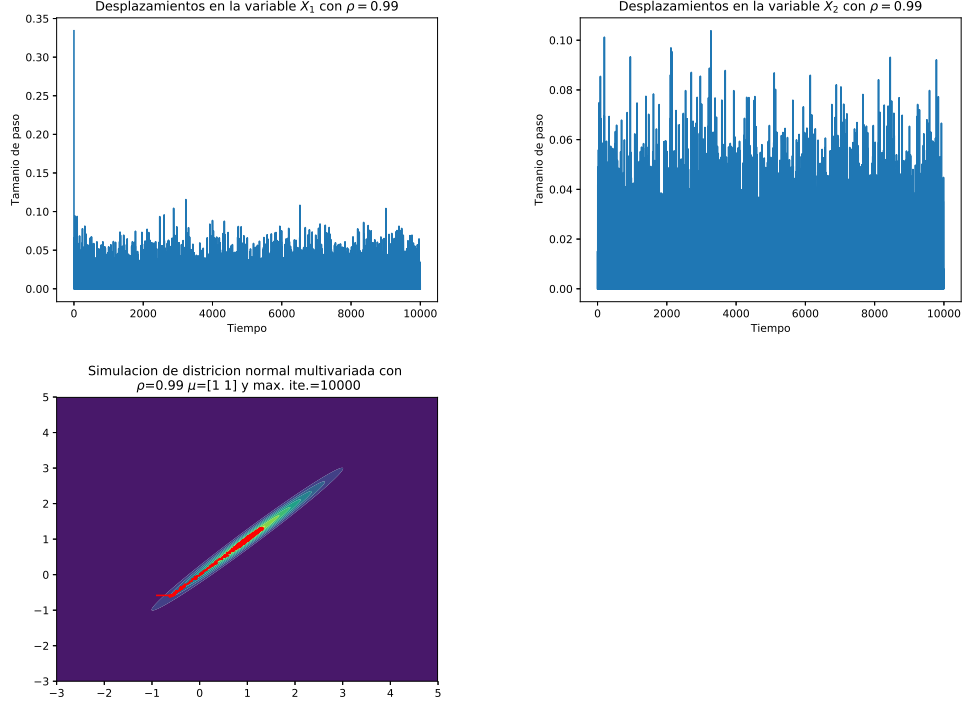


Figura 2: Desplazamientos realizados por la cadena de markov en la parte superior y el recorrido en la parte inferior.

## 2. Simulación distribución Weibull

Considere los tiempos de falla  $t_1, \dots, t_n$  con distribución de Weibull( $\alpha, \lambda$ ):

$$f(t_i|\alpha, \lambda) = \alpha \lambda t_i^{\alpha-1} e^{-t_i^\alpha \lambda} \quad (6)$$

Se asumen como a priori  $\alpha \sim \exp(c)$  y  $\lambda|\alpha \sim \text{Gamma}(a, b)$ , por lo tanto,  $f(\alpha, \lambda) = f(\lambda|\alpha)f(\alpha)$ . Así, para la distribución posterior se tiene:

$$f(\alpha, \lambda|\hat{t}) \propto f(\hat{t}|\alpha, \lambda)f(\alpha, \lambda) \quad (7)$$

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

■ Propuesta 1:

$$\lambda_p|\alpha, \hat{t} \sim \text{Gamma}(\alpha + b, b + \sum_{i=1}^n t_i^\alpha) \quad (8)$$

dejando  $\alpha$  fijo.

- Propuesta 2:

$$\alpha_p | \lambda, \hat{t} \sim \text{Gamma}(n + 1, -\log(b) - \log(r_1) + c) \quad (9)$$

con  $r_i = \prod_{i=1}^n t_i$ , dejando  $\lambda$  fijo.

- Propuesta 3:

$$\alpha_p \sim \exp(c) \lambda_p | \alpha_p \sim \text{Gamma}(\alpha_p, b) \quad (10)$$

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

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

### Comentarios

En este problema ampliamente explicado en [2] se simula la función posterior por medio del conjugado de dos apriori y la verosimilitud de los datos. Por lo tanto función verosimilitud es:

$$l(\text{data} | \alpha, \lambda) = f(\hat{t} | \alpha, \lambda) \propto \alpha^n \lambda^n \prod_{i=1}^n t_i^{\alpha-1} e^{-\lambda \sum_{i=1}^n t_i^\alpha} \quad (11)$$

Las funciones apriori se indican a continuación ( $\alpha = a$ ):

$$f(\lambda | \alpha) f(\alpha) \propto \left( \frac{b^a}{\Gamma(a)} \lambda^{a-1} e^{-b\lambda} \right) c e^{-\alpha c} \quad (12)$$

Entonces la distribución aposteriori resulta como el producto:

$$f(\alpha, \lambda | \hat{t}) \propto \left( \frac{b^a}{\Gamma(a)} \lambda^{a-1} e^{-b\lambda} \right) c e^{-\alpha c} \left( \alpha^n \lambda^n \prod_{i=1}^n t_i^{\alpha-1} e^{-\lambda \sum_{i=1}^n t_i^\alpha} \right) \quad (13)$$

Es importante mencionar que en algunos casos las propuestas podían generar valores que no estuvieran en el soporte, esos casos se ignoraron. En la implementación se consideran todos los kernels con los mismos pesos, el burn-in que se dió es de 100 iteraciones y el número máximo de iteraciones fue de 1000. En promedio los valores que se obtuvieron fueron  $\alpha = 0,90636827$  y  $\beta = 1,16923842$ .

En la figura 3 se presentan varios resultados interesantes, en la figura de la parte superior izquierda se realiza una estimación de las funciones de densidad tanto de  $\alpha$  como de  $\beta$ . En la parte superior derecha, se grafica el mapa de contorno de la función posteriori y los estados visitados por la cadena de markov. En la segunda fila se grafican los estados visitados por  $\alpha$  (parte inferior izquierda) y  $\beta$  (parte inferior derecha).

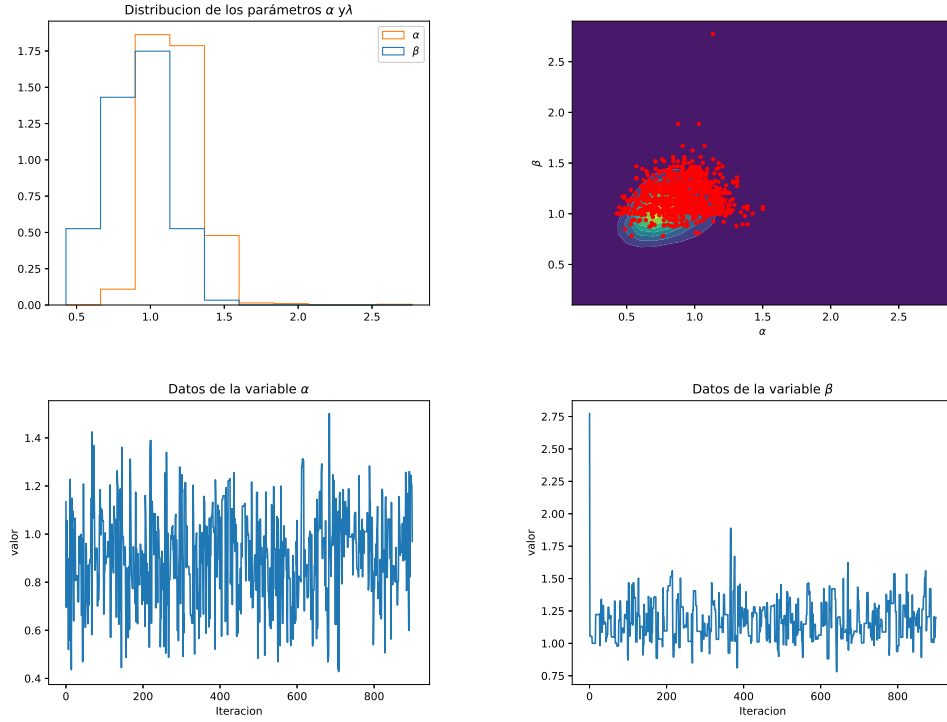


Figura 3: Desplazamientos realizados por la cadena de markov en la parte superior y el recorrido en la parte inferior.

### 3. Problema de las bombas de agua en la central nuclear

Considere el ejemplo referente al número de fallas de bombas de agua en una central nuclear [1] [3], 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  $\lambda_i$  son independientes entre si), con distribuciones a priori

$\lambda_i|\beta \sim \text{Gamma}(\alpha, \beta)$  y  $\beta \sim \text{Gamma}(\gamma, \delta)$ , por lo tanto:

$$f(\lambda_1, \dots, \lambda_n, \beta) = f(\lambda_1|\beta) \dots f(\lambda_n|\beta) f(\beta) \quad (14)$$

Para la distribución posterior se tiene

$$f(\lambda_1, \dots, \lambda_n, \beta|\hat{p}) \propto L(\hat{p}, \hat{\lambda}, \beta) f(\lambda_1, \dots, \lambda_n, \beta) \quad (15)$$

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

- $\lambda_1|\hat{\lambda}_i, \beta, \hat{y} \sim \text{Gamma}(p_i + \alpha, t_i + \beta)$
- $\beta|\hat{\lambda}, \hat{t} \sim \text{Gamma}(n\alpha + \gamma, \delta + \sum_{i=1}^n \lambda_i)$

verifique que estas son propuestas de Gibbs. Utilizar la siguiente tabla con los parámetros a priori  $\alpha = 1,8$ ,  $\gamma = 0,01$  y  $\delta = 1$ .

<b>Pump</b>	1	2	3	4	5	6	7	8	9	10
<b>Failures</b>	5	1	5	14	3	19	1	1	4	22
<b>Time</b>	94.32	15.72	62.88	125.76	5.24	31.44	1.05	1.05	2.10	10.48

### Comentarios

Dada las distribuciones a priori asociadas  $\lambda_i \sim \text{Gamma}(\alpha, \beta)$ ,  $\beta \sim \text{Gamma}(\gamma, \delta)$ . La distribución conjunta es entonces:

$$\begin{aligned} & \pi(\lambda_1, \dots, \lambda_{10}, \beta|t_1, \dots, t_{10}, p_1, \dots, p_{10}) \\ & \propto \prod_{i=1}^{10} \{(\lambda_i t_i)^{p_i} e^{-\lambda_i t_i} \lambda_i^{\alpha-1} e^{-\beta \lambda_i}\} \beta^{10\alpha} \beta^{\gamma-1} e^{-\delta \beta} \\ & \propto \prod_{i=1}^{10} \{\lambda_i^{p_i+\alpha-1} e^{-(t_i+\beta)\lambda_i}\} \beta^{10\alpha+\gamma-1} e^{-\delta \beta} \end{aligned} \quad (16)$$

y siguiendo una descomposición natural (esta descomposición refleja la estructura jerárquica del modelo) de  $\pi$  en las distribuciones condicionales se tiene lo siguiente:

$$\begin{aligned} & \lambda_1|\hat{\lambda}_i, \beta, \hat{y} \sim \text{Gamma}(p_i + \alpha, t_i + \beta) \\ & \beta|\hat{\lambda}, \hat{t} \sim \text{Gamma}(n\alpha + \gamma, \delta + \sum_{i=1}^n \lambda_i) \end{aligned} \quad (17)$$

	$\beta$	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	$\lambda_6$	$\lambda_7$	$\lambda_8$	$\lambda_9$	$\lambda_{10}$
<b>Resultados obtenido</b>	2.522	0.068	0.149	0.106	0.122	0.6072	0.6185	0.829	0.822	1.2733	1.827
<b>Resultados de la tesis</b>	-	0.059	0.102	0.089	0.116	0.116	0.609	0.893	0.881	1.584	1.992

Por lo tanto queda verificado que estas propuestas son Gibbs, pues muestreando de las distribuciones condicionales es posible simular la distribución posteriori.

Esta simulación se programó considerando un burn-in de 1000 iteraciones y un número total de iteraciones de 10000. En la siguiente tabla se muestra el promedio obtenido de los estados visitados por la cadena de markov (sin considerar el burn-in). Se menciona que los tanto la tabla como las propuestas están distintas en el libro [1] de que en las instrucciones de la tarea.

Los resultados obtenidos no fueron exactamene a los obtenidos en [3] ( $\beta = 2,5$ ), la razón de esto es que hay datos distintos entre el libro [1] pag. 385, no obstante si se obtuvieron resultados bastante similares a los del trabajo de tesis *Experimentación con un algoritmo de MCMC multiescala y autoajustable* por Patricia Bautista Otero. Los resultados de referencia de la tesis se agregan en la fila inferior.

En la figura 4 se presentan los estados visitados por la cadena de markov en cada una de las variables de  $\lambda_i$ , principalmente se observa todas las variables alcanzaron como máximo un valor de 3. En la figura 5 se ilustran los estados (valores) obtenidos por la cadena de markov en la variable  $\beta$ , se observa que aunque inicialmente tiene valores cercanos a 25, tiene una convergencia a valores menores que 5, y su comportamiento tiende a estabilizarse.



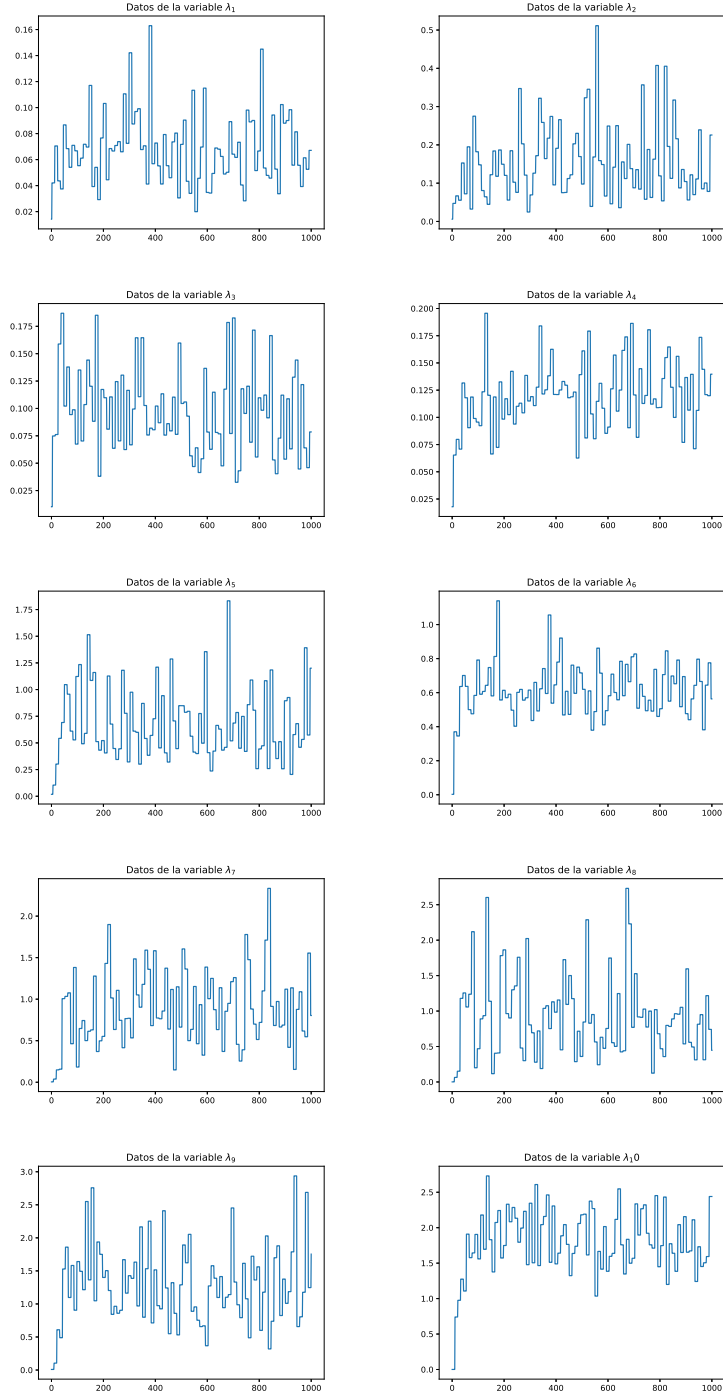


Figura 4: Estados visitados por la cadena de markov, en el eje de las abscisas se indican las iteraciones y en el de las ordenadas el valor de cada variable.

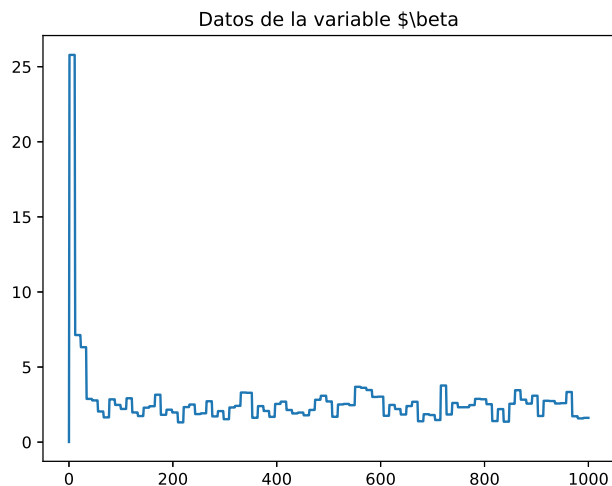


Figura 5: Estados visitados por la cadena de markov, en el eje de las abscisas se indican las iteraciones y en el de las ordenadas el valor de cada variable.

## 1 Referencias

- 2 [1] C. P. Robert, G. Casella, G. Casella, Introducing monte carlo methods  
3 with r, volume 18, Springer, 2010.
- 4 [2] D. Kundu, Bayesian inference and life testing plan for the weibull dis-  
5 tribution in presence of progressive censoring, Technometrics 50 (2008)  
6 144–154.
- 7 [3] R. A. Norton, J. A. Christen, C. Fox, Sampling hyperparameters in hie-  
8 rarchical models: Improving on gibbs for high-dimensional latent fields  
9 and large datasets, Communications in Statistics-Simulation and Compu-  
10 tation 47 (2018) 2639–2655.