

Algoritmos MCMC

Prof. Caio Azevedo

Exemplo da normal bivariada

- Seja $\mathbf{Y} = (Y_1, Y_2) \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Desejamos estimar as fdp's marginais de Y_1, Y_2 , com $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$, conhecidos.
- Sabemos que $Y_i \sim N(\mu_i, \sigma_i^2), i = 1, 2$.
- Suponha, no entanto, que saibamos apenas que:

$Y_1|y_2, \boldsymbol{\mu}, \boldsymbol{\Sigma} \sim N_1(\bar{\mu}_1, \bar{\sigma}_1^2)$, em que $\bar{\mu}_1 = \mu_1 + \sigma_{12} (\sigma_2^2)^{-1} (y_2 - \mu_2)$ e $\bar{\sigma}_1^2 = \sigma_1^2 - (\sigma_{12})^2 (\sigma_2^2)^{-1}$.

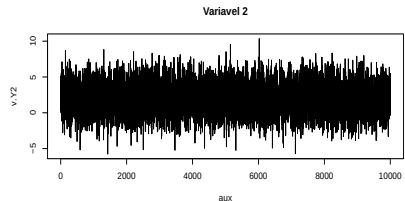
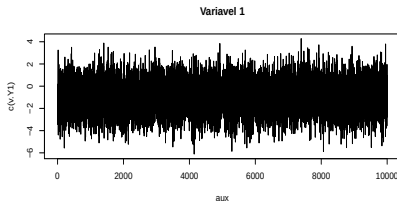
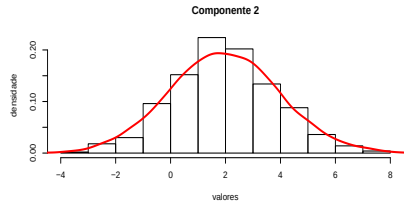
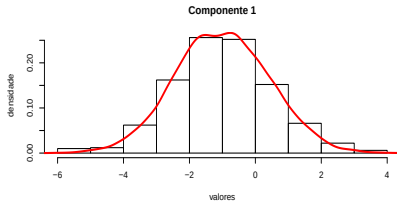
$Y_2|y_1, \boldsymbol{\mu}, \boldsymbol{\Sigma} \sim N_1(\bar{\mu}_2, \bar{\sigma}_2^2)$, em que $\bar{\mu}_2 = \mu_2 + \sigma_{12} (\sigma_1^2)^{-1} (y_1 - \mu_1)$ e $\bar{\sigma}_2^2 = \sigma_2^2 - (\sigma_{12})^2 (\sigma_1^2)^{-1}$.

- As distribuições condicionais podem ser utilizadas para estimar as distribuições marginais?

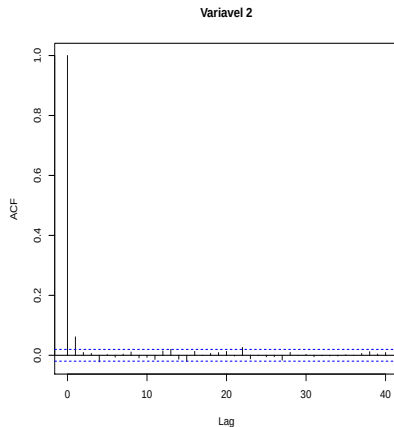
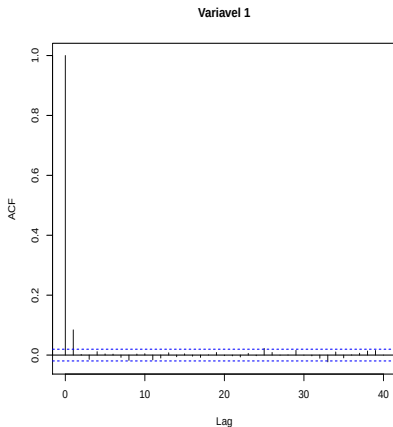
Simulação iterativa

- Simule $y_1^{(1)} \sim N(0, 1)$, por exemplo. Com este valor, simule $y_2^{(1)} | y_1^{(1)}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$.
- Repita o passo acima R vezes, obtendo-se $(y_1^{(1)}, y_2^{(1)}), (y_1^{(2)}, y_2^{(2)}), \dots, (y_1^{(R)}, y_2^{(R)})$

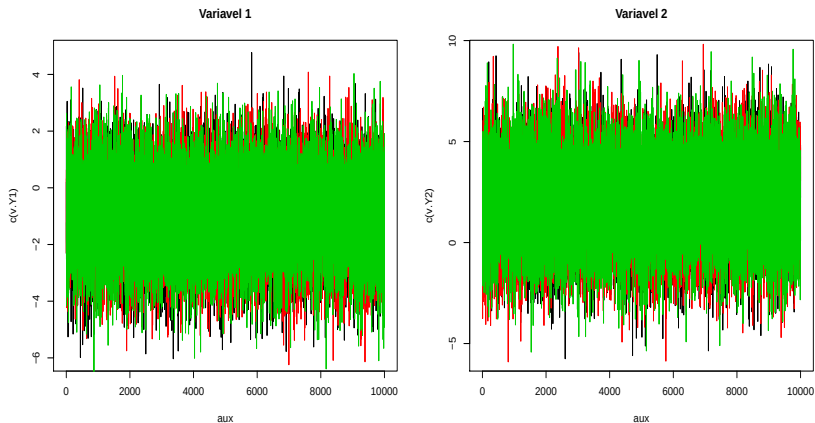
Histograma das amostras MCMC



ACF das amostras MCMC



Trace plots das amostras MCMC



Cadeias de Markov

- Estudo de sequências (conjuntos) de variáveis aleatórias que guardam alguma estrutura dependência entre si.
- Exemplo: seja $X_1, X_2, \dots$ um sequência de Bernoullis que representam resultados de lançamentos de duas moedas (0, cara ; 1, coroa).
- Caso $X_i = 0$, lança-se uma moeda com probabilidade de cara igual a 0,55, caso contrário lança-se uma moeda com probabilidade de cara igual à 0,35.

Exemplo: lançamento de moedas

- Matriz de transição

$$\mathbf{P} = \begin{bmatrix} P(X_i = 0 | x_{i-1} = 0) & P(X_i = 1 | x_{i-1} = 0) \\ P(X_i = 0 | x_{i-1} = 1) & P(X_i = 1 | x_{i-1} = 1) \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} 0,55 & 0,45 \\ 0,35 & 0,65 \end{bmatrix}$$

- Distribuição estacionária: $\mathbf{x} = \mathbf{P}\mathbf{x}$.

Exemplo: lançamento de moedas

- Resultado: Se a Cadeia de Markov (definida pela matriz de transição), for reversível, a distribuição estacionária existe e é única.
- Obtida através de observações da cadeia de Markov, $\mathbf{x} = \mathbf{P}^n$, para n suficiente grande.
- Neste caso $\mathbf{x} = [0, 45; 0, 55]$

Exemplo: normal bivariado

- No exemplo da normal bivariada, a matriz de transição (que possui um número infinito não enumerável de elementos) é definida pelas densidades de transição (distribuições condicionais completas):

$$Y_1|y_2, \boldsymbol{\mu}, \boldsymbol{\Sigma}; Y_2|y_1, \boldsymbol{\mu}, \boldsymbol{\Sigma}$$

- Neste caso, faz sentido calcular $P(Y_1 \in a|y_2, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ e $P(Y_2 \in b|y_1, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, em que a, b são intervalos da reta.
- Resultado: se a Cadeia de Markov, com matriz de transição definida pelas condicionais completas for reversível, sua distribuição estacionária será única e convergirá para as distribuições marginais de interesse.

Algoritmos de Monte Carlo via Cadeias de Markov

- O conjunto de algoritmos que permitem simular variáveis (iterativamente) a partir das distribuições condicionais completas é conhecido como algoritmos de Monte Carlo via Cadeias de Markov (MCMC).
- Monte Carlo (resolver integrais) Cadeias de Markov (gera cadeias de Markov).
- Aplicações: obter distribuições (marginais, condicionais, conjuntas) de interesse. Em particular: Inferência Bayesiana.

Aplicações em Inferência Bayesiana

- Distribuição inversa gama $X \sim IG(r, \gamma)$:

$$p(x) = \frac{\gamma^r}{\Gamma(r)} e^{-x/\gamma} x^{-(r+1)} \mathbb{1}_{(0, \infty)}(x)$$

- Distribuição normal-inversa gama $(X, Y) \sim NIG(\mu, \nu, r, \gamma)$

$$X|y \sim N(\mu, y/\nu)$$

$$Y \sim IG(r, \gamma)$$

Continuação

■ Densidade conjunta

$$\begin{aligned}
 p(x, y) &= \frac{\sqrt{\nu}}{\sqrt{2\pi}} y^{-1/2} y^{-(r+1)} \exp\left(-\frac{\gamma}{y}\right) \exp\left\{-\frac{1}{2} \frac{\nu(x-\mu)^2}{\sigma^2}\right\} \\
 &\times \mathbb{1}_{(0,\infty)}(y) \mathbb{1}_{(-\infty,\infty)}(x)
 \end{aligned}$$

■ Distribuição marginal de X , $X \sim t_{(2r)}\left(\mu, \sqrt{\frac{\gamma}{r\nu}}\right)$,

$$\begin{aligned}
 p(x) &= \frac{\Gamma\left(\frac{2r+1}{2}\right)}{\Gamma\left(\frac{2r}{2}\right)\Gamma\left(\frac{1}{2}\right)} (\sqrt{2r}\delta)^{-1} \left[1 + \frac{(x-\mu)^2}{2r\delta^2}\right]^{-\frac{2r+1}{2}} \mathbb{1}_{(-\infty,\infty)}(x) \\
 \delta^2 &= \sqrt{\frac{\gamma}{r\nu}}
 \end{aligned}$$

Aplicações em Inferência Bayesiana

- Seja $X_1|\boldsymbol{\theta}, \dots, X_n|\boldsymbol{\theta}, \boldsymbol{\theta} = (\mu, \sigma^2)$ uma amostra aleatória de $X|\boldsymbol{\theta} \sim N(\mu, \sigma^2)$.
- Família conjugada (normal inversa gama)

$$\mu|\sigma^2 \sim N(\alpha, \sigma^2/\kappa)$$

$$\sigma^2 \sim \text{IG}(\gamma, \beta)$$

Continuação

■ Posteriori conjunta

$$\mu|\mathbf{x}, \sigma^2 \sim N(\alpha^*, \sigma^2/(\nu^*))$$

$$\sigma^2|\mathbf{x} \sim \text{IG}(r^*, \gamma^*)$$

em que $\nu^* = \nu + n$, $\gamma^* = \frac{1}{2} \left[\frac{n\nu}{n+\nu} (\bar{X} - \alpha)^2 + (n-1)s^2 + \gamma \right]$,

$$r^* = \frac{n}{2} + r.$$

■ Além disso, $\mu|\mathbf{x} \sim t_{(2r^*)} \left(\alpha^*, \sqrt{\frac{\gamma^*}{r^*\nu^*}} \right).$

Continuação

- Pode-se provar que (distribuições condicionais completas):

$$\mu|\mathbf{x}, \sigma^2 \sim N(\alpha^*, \sigma^2/(\nu^*))$$

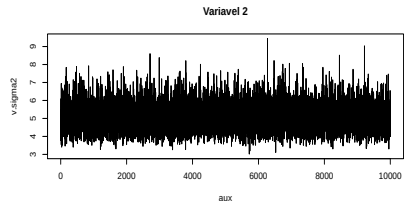
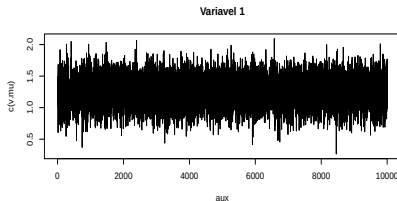
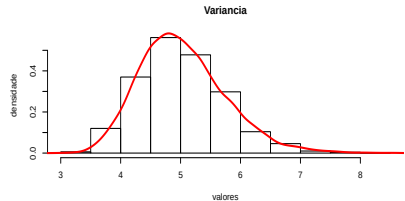
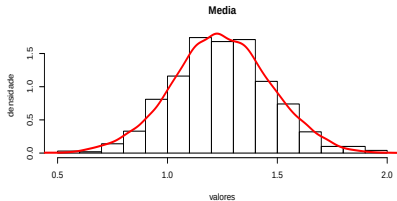
$$\sigma^2|\mathbf{x}, \mu \sim \text{IG}(r^{**}, \gamma^{**})$$

em que

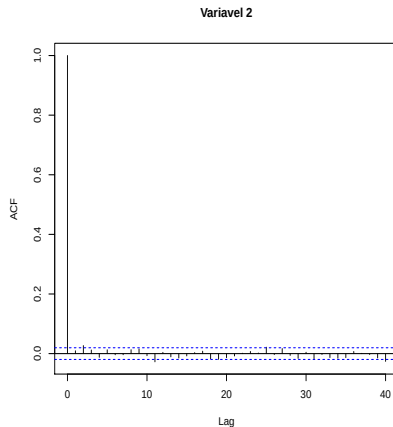
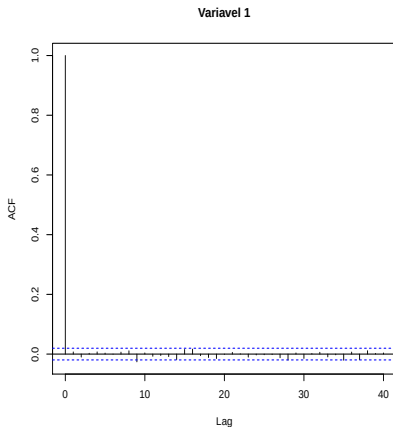
$$r^{**} = \frac{n+1}{2} + r; \gamma^{**} = \frac{n}{2} (\bar{x} - \mu)^2 + \frac{\nu}{2} (\mu - \alpha)^2 + \frac{1}{2}(n-1)s^2 + \gamma$$

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i; s^2 = \frac{1}{n-2} \sum_{i=1}^n (x_i - \bar{x})^2.$$

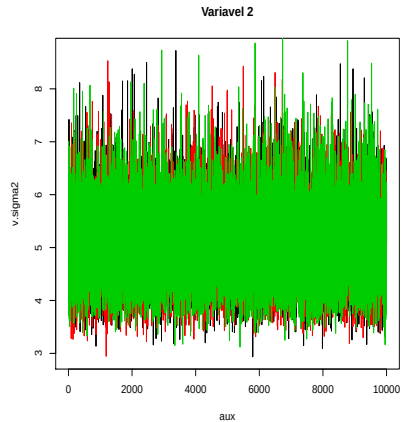
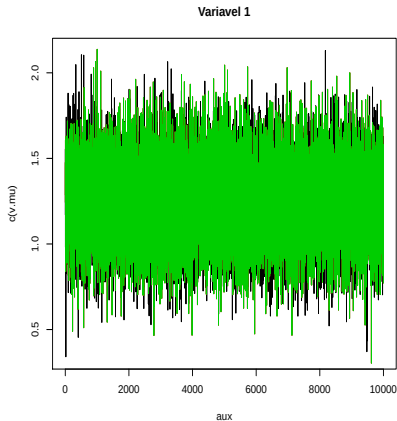
Histograma das amostras MCMC



ACF das amostras MCMC



Trace plots das amostras MCMC



Definição geral dos algoritmos MCMC

- Seja $p(\boldsymbol{\theta}|\mathbf{x}) \propto p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ a distribuição a posteriori de interesse (com forma analítica intratável).
- Considere a seguinte partição para o (vetor) p-paramétrico $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_k), k \leq p$.
- Sejam $\boldsymbol{\theta}_j|\boldsymbol{\theta}_{(-j)}, \mathbf{x}, i = 1, 2, \dots, k$ em que $\boldsymbol{\theta}_{(-j)}$ denota o vetor paramétrico $\boldsymbol{\theta}$ menos a componente $\boldsymbol{\theta}_j$, as **distribuições condicionais completas**.

Definição geral dos algoritmos MCMC

- Inicie a cadeia com um conjunto de valores iniciais apropriados.
- Para $r = 1, 2, \dots, R$, $j = 1, 2, \dots, k$ simule $\theta_j^{(r)}$ de $\theta_j | \theta_{(-j)}^{(r-1)}, \mathbf{x}$.
- A partir de algum $B > 1$ (burn-in), retenha os valores a cada t (espaçamento) iterações.
- Os valores retidos, se a cadeia tiver alcançado a convergência, corresponderá à uma amostra aleatória de das posteriores (distribuições) de interesse.

Questões de interesse

- 1 Como simular de θ_j caso a distribuição condicional completa não seja conhecida ou possível de simular diretamente?
- 2 Como verificar a convergência da cadeia gerada?
- 3 Como determinar B, t, R ?

Simulação das distribuições condicionais completas

- Se todas as distribuições condicionais completas forem conhecidas e fáceis de simular, teremos o algoritmo do amostrador de Gibbs (“Gibbs sampling”).
- Como no exemplo anterior da distribuição normal.
- Ponto importante, identificada a distribuição condicional completa, eg, normal, inversa gama, t de Student, devemos escolher algoritmos apropriados para simular delas (eg, transformada inversa, rejeição, rejeição adaptativa, amostragem por importância).

Amostrador de Gibbs

- Inicie as cadeias com valores iniciais conveniente.

- Simule, para $r=1,2,\dots,R$:

1 $\theta_1^{(r)}$ de $\theta_1^{(r)} | \theta_2^{(r-1)}, \theta_3^{(r-1)}, \dots, \theta_k^{(r-1)}, \mathbf{x}$.

2 $\theta_2^{(r)}$ de $\theta_2^{(r)} | \theta_1^{(r)}, \theta_3^{(r-1)}, \dots, \theta_k^{(r-1)}, \mathbf{x}$.

3 $\theta_3^{(r)}$ de $\theta_3^{(r)} | \theta_1^{(r)}, \theta_2^{(r)}, \dots, \theta_k^{(r-1)}, \mathbf{x}$.

⋮

4 $\theta_k^{(r)}$ de $\theta_k^{(r)} | \theta_1^{(r)}, \theta_2^{(r)}, \dots, \theta_{k-1}^{(r)}, \mathbf{x}$.

Regressão Probit

- Seja $Y_1, \dots, Y_n$ uma a.a. de $Y \sim \text{Bernoulli}(p_i)$, $p_i = \Phi(\mathbf{X}'_i\beta)$.
Desejamos estimar $\beta = (\beta_0, \beta_1, \dots, \beta_p)$.
- Verossimilhança:

$$L(\beta) = \prod_{i=1}^n p_i^{y_i} (1 - p_i)^{1-y_i}$$

- Defina:

$$Y_i = I_{(Z_i > 0)}$$

$$Z_i \sim N(\mathbf{X}'_i\beta, 1)$$

Cont. Regressão Probito

- Assim, pode-se provar que (distribuição condicional completa)

$$Z_i|y_i, \mathbf{X}_i, \boldsymbol{\beta} = \begin{cases} N_{(0,\infty)}(\beta_0 + \mathbf{X}_i\boldsymbol{\beta}_1, 1) & \text{se } y_i = 1 \\ N_{(-\infty,0)}(\beta_0 + \mathbf{X}_i\boldsymbol{\beta}_1, 1) & \text{se } y_i = 0 \end{cases} \quad (1)$$

- Assim, pode-se trabalhar com a seguinte verossimilhança

$$L(\boldsymbol{\beta}, \mathbf{z}, \mathbf{y}) \propto \prod_{i=1}^n \exp \left[-\frac{1}{2} (\mathbf{z}_i - \mathbf{X}_i\boldsymbol{\beta})' (\mathbf{z}_i - \mathbf{X}_i\boldsymbol{\beta}) \right]$$

Cont. Regressão Probit

- Assuma que

$$p(\beta) \propto \exp \left[-\frac{1}{2} (\beta - \mu_\beta)' \Psi_\beta^{-1} (\beta - \mu_\beta) \right]$$

- Assim, a distribuição condicional completa do vetor β , será

$$\beta | \mathbf{y}, \mathbf{x}, \mathbf{z} \sim N_{p+1}(\hat{\Psi}_\beta \hat{\mu}_\beta, \hat{\Psi}_\beta),$$

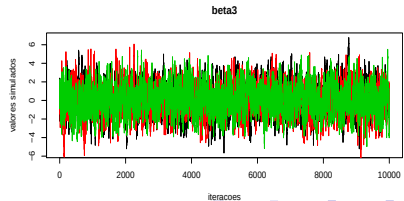
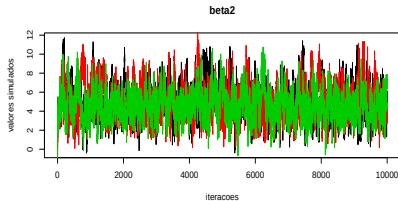
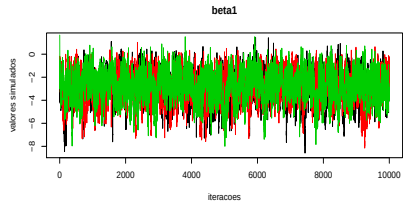
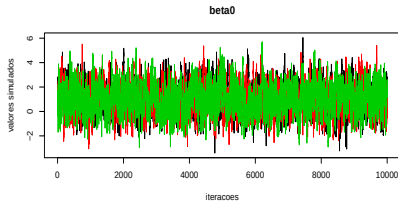
em que

$$\hat{\Psi}_\beta = [\mathbf{X}'\mathbf{X} + \Psi_\beta^{-1}]^{-1}, \hat{\mu}_\beta = \mathbf{X}'\mathbf{z} + \Psi_\beta \mu_\beta$$

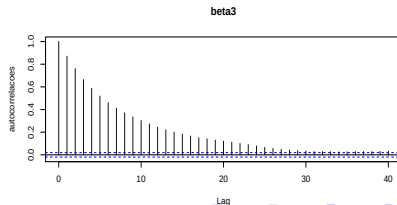
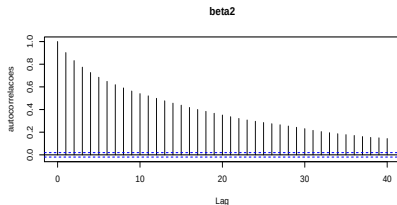
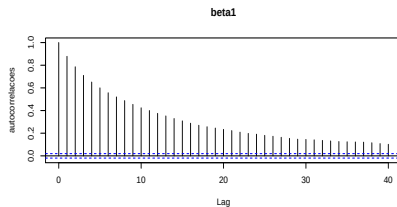
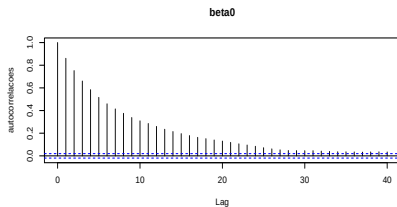
Cont. Regressão Probit: Algoritmo de Gibbs

- 1 Inicie a cadeia com valores iniciais adequados para β .
- 2 Simule Z_i de $Z_i|\mathbf{X}_i, y_i, \beta$, $i=1, \dots, n$ de modo independente.
- 3 Simule β de $\beta|\mathbf{x}, \mathbf{y}, \mathbf{z}$.
- 4 Repita os passos 2 e 3 acima R vezes.

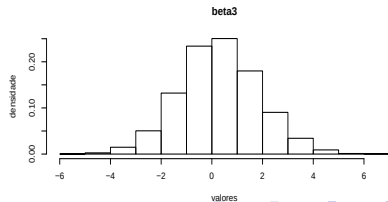
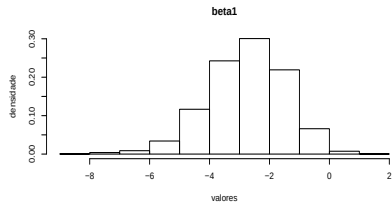
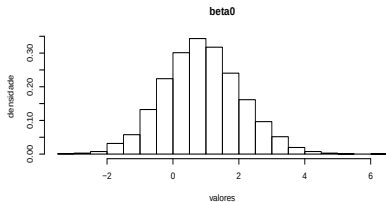
Trace plots das amostras MCMC



ACF das amostras MCMC



Histogramas das amostras MCMC



Como verificar a convergência das cadeias geradas

- Assuma que a distribuição a posteriori exista.
- A convergência não pode depender dos estados iniciais (valores iniciais) das cadeias.
- As cadeias não podem ficar “presas” em subconjuntos dos espaços paramétricos.
- Inspeção visual: monitoramento visual dos valores simulados ou de estatísticas de convergência.
- Estatísticas de convergência: Geweke e Gelman-Rubin.

Procedimento de Geweke

- Consiste em subdividir uma única cadeia (para cada parâmetro) em várias sub cadeias disjuntas e consecutivas e aplicar testes para igualdade de médias (supondo variância desiguais) através do teste t.
- Sejam $C_1, C - 2, \dots, C_M$, as M sub cadeias e $\bar{x}_k, s_m^2, n_m$ a média amostral, variância amostral e tamanho da cadeia m.
- Estatística do teste
$$z_{ij} = \frac{\bar{x}_i - \bar{x}_j}{\sqrt{\frac{s_i^2}{n_i} + \frac{s_j^2}{n_j}}}.$$

Procedimento de Gelman Rubin

- Consiste em uma análise de variância considerando M diferentes cadeias para cada parâmetro.
- Defina

$$W = \frac{1}{M} \sum_{j=1}^M s_j^2; s_j^2 = \frac{1}{R-1} \sum_{i=1}^R (\theta_{ij} - \bar{\theta}_j)^2$$

$$B = \frac{R}{M-1} \sum_{j=1}^M (\bar{\theta}_j - \bar{\bar{\theta}})^2; \bar{\bar{\theta}} = \frac{1}{M} \sum_{j=1}^M \bar{\theta}_j$$

$$\widehat{Var}(\theta) = \left(1 - \frac{1}{R}\right) W + \frac{1}{R} B; R = \frac{\widehat{Var}(\theta)}{W}$$

Suporte computacional: pacote CODA (no R)

- Faz gráficos dos valores simulados (trace plots), de auto correlações, de medidas descritivas ergódicas, de estatística de ajuste, auto-correlações.
- Calcula estimativas Bayesianas pontuais e intervalares.
- Uma única amostra de cada parâmetro: matriz em que as linhas são os valores simulados e as colunas são os parâmetros. Transforma num objeto *mcmc*.
- Várias amostras de cada parâmetro: concatenar verticalmente objetos *mcmc* num objeto *mcmc.list*.

Medidas descritivas: amostra completa

Sem título

```
Iterations = 1:10001
Thinning interval = 1
Number of chains = 1
Sample size per chain = 10001
```

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naïve SE	Time-series SE
[1,]	0.9722	1.173	0.01173	0.04995
[2,]	-2.8660	1.352	0.01351	0.07916
[3,]	4.9517	1.797	0.01797	0.12748
[4,]	0.1163	1.602	0.01602	0.06929

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
var1	-1.252	0.1707	0.93942	1.749	3.3430
var2	-5.739	-3.7441	-2.77690	-1.902	-0.4978
var3	1.803	3.6983	4.82238	6.078	8.9307
var4	-3.075	-0.9409	0.08717	1.190	3.2710

Medidas descritivas: burn-in

Sem título

```
Iterations = 1:5001
Thinning interval = 1
Number of chains = 1
Sample size per chain = 5001
```

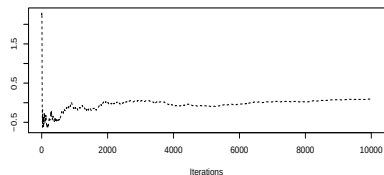
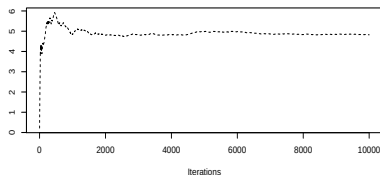
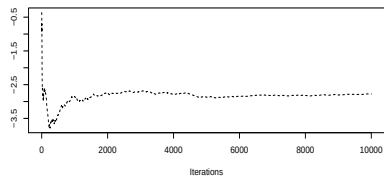
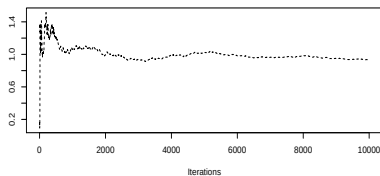
1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naïve SE	Time-series SE
[1,]	0.8815	1.183	0.01673	0.07858
[2,]	-2.7625	1.295	0.01831	0.09585
[3,]	4.7979	1.698	0.02401	0.14580
[4,]	0.2892	1.574	0.02226	0.09981

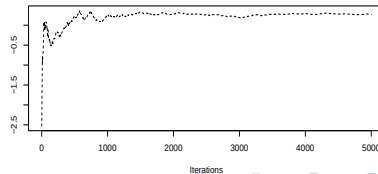
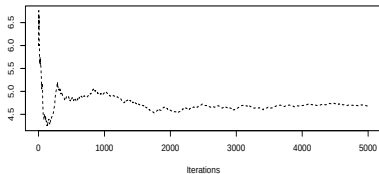
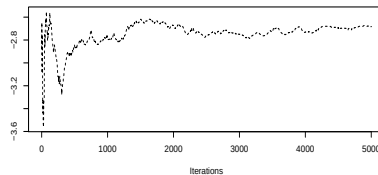
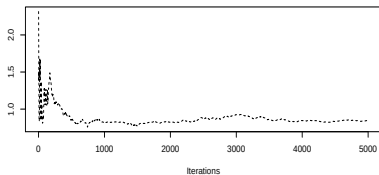
2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
var1	-1.408	0.0793	0.8514	1.660	3.2292
var2	-5.457	-3.5895	-2.6828	-1.855	-0.4636
var3	1.791	3.6326	4.6810	5.849	8.4668
var4	-2.792	-0.7579	0.2698	1.319	3.4214

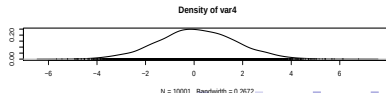
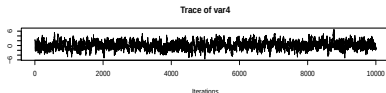
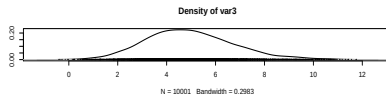
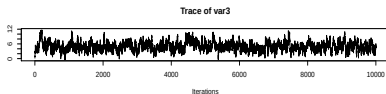
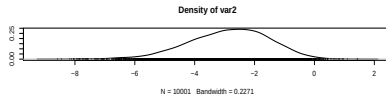
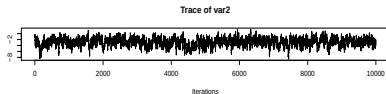
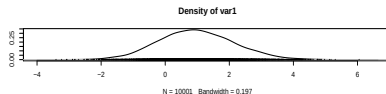
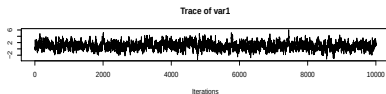
Cumulative plots: amostra completa



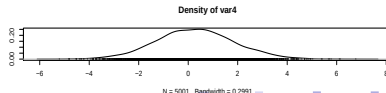
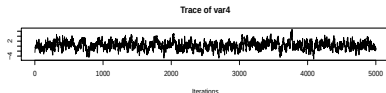
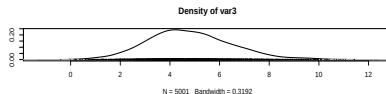
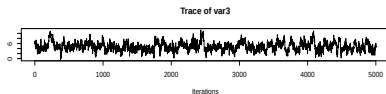
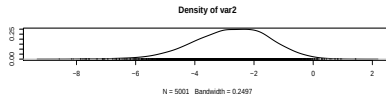
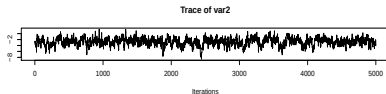
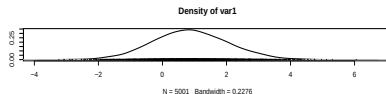
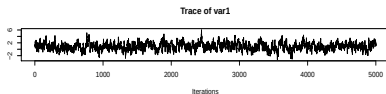
Cumulative plots: burn in



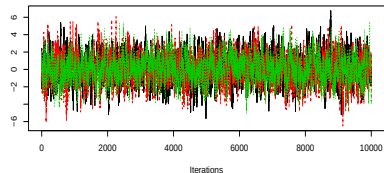
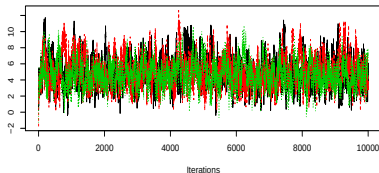
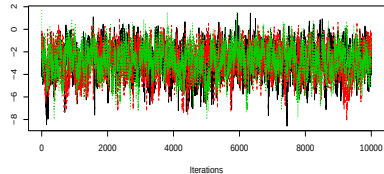
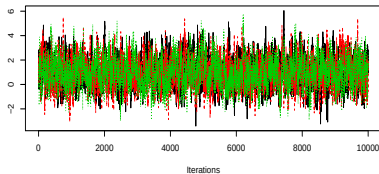
Trace plots: amostra completa



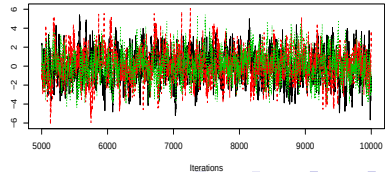
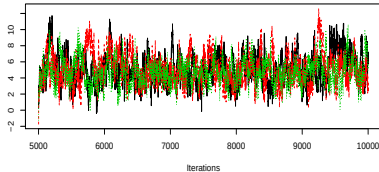
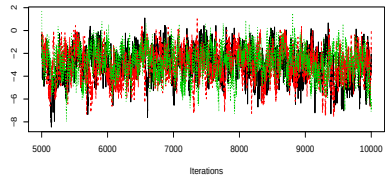
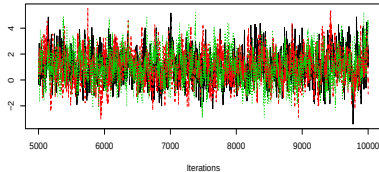
Trace plots: burn in



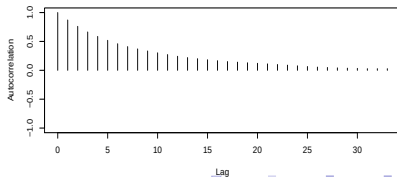
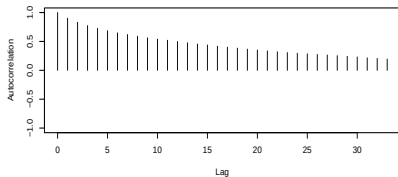
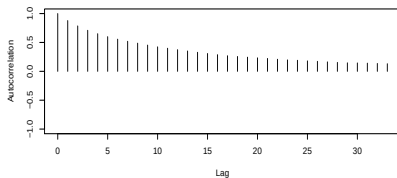
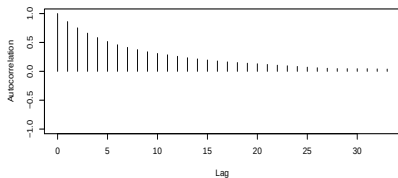
Trace plots: amostra completa



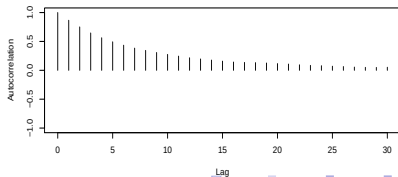
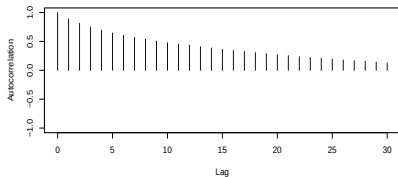
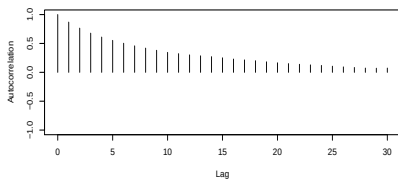
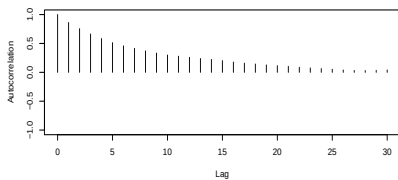
Trace plots: burn in



ACF: amostra completa



ACF: burn in



Estatística de Geweke

Sem título

Cadeia Completa

Fraction in 1st window = 0.5
Fraction in 2nd window = 0.5

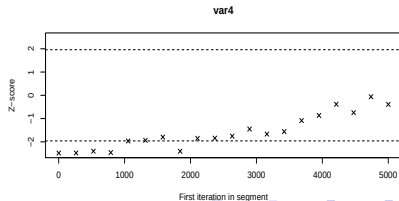
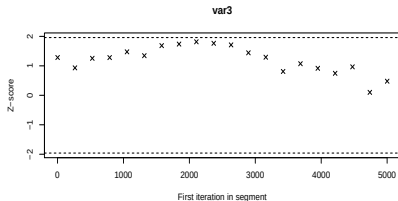
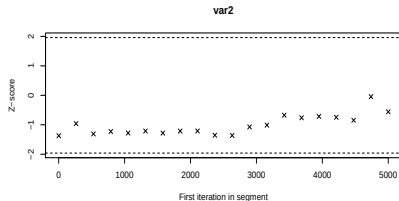
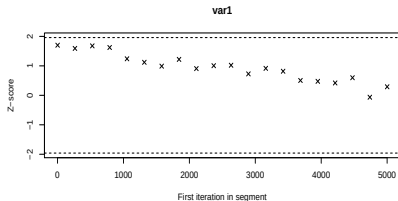
var1	var2	var3	var4
1.701	-1.372	1.283	-2.483

Bur-in

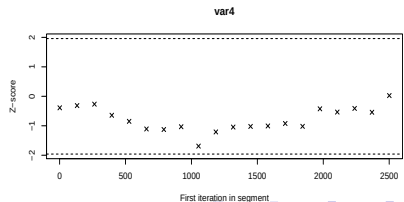
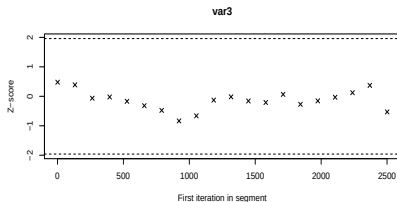
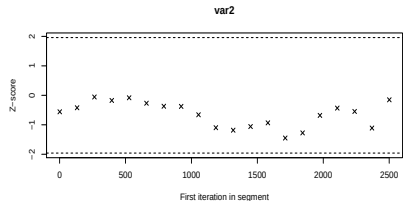
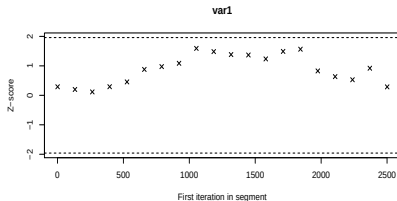
Fraction in 1st window = 0.5
Fraction in 2nd window = 0.5

var1	var2	var3	var4
0.2903	-0.5586	0.4784	-0.3899

Gráficos das estatística de Geweke: amostra completa



Gráficos das estatística de Geweke: burn in



Estatística de Gelman Rubin

Sem título

Cadeia completa

Potential scale reduction factors:

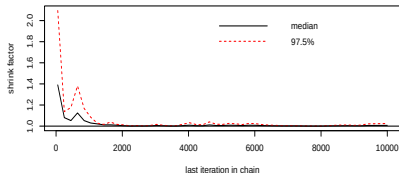
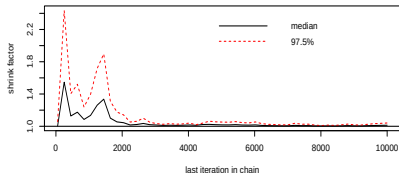
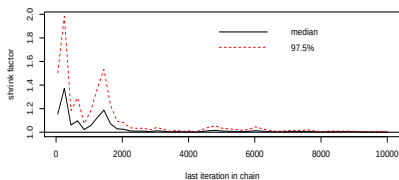
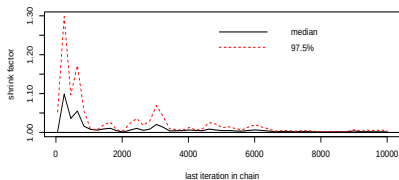
	Point est.	97.5% quantile
[1,]	1.00	1.01
[2,]	1.00	1.00
[3,]	1.01	1.04
[4,]	1.01	1.02

Burn in

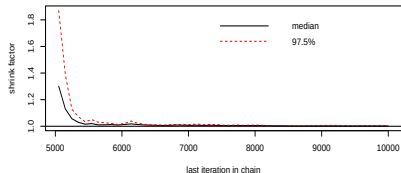
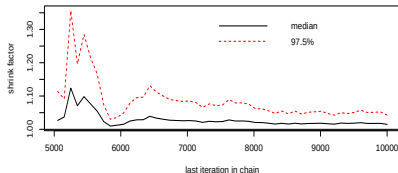
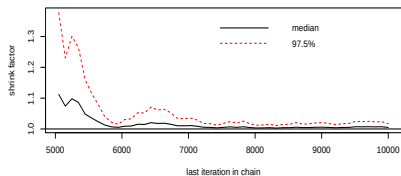
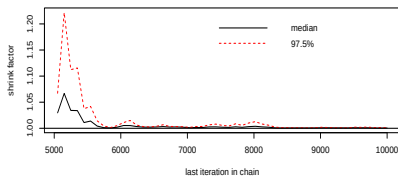
Potential scale reduction factors:

	Point est.	97.5% quantile
[1,]	1.00	1.00
[2,]	1.01	1.02
[3,]	1.01	1.04
[4,]	1.00	1.00

Gráficos das estatística de Gelman Rubin: a. completa



Gráficos das estatística de Gelman Rubin: burn in



Cont.

- O que fazer quando não conhecemos a forma das distribuições condicionais completas (no sentido de que não a reconhecemos como uma distribuição conhecida)?
- Tem-se que se utilizar algum algoritmo auxiliar para simular dessa densidade: Metropolis-Hastings (Metropolis-Hastings dentro do amostrador de Gibbs), rejeição adaptativa, amostragem por importância, amostragem por corte “slice sampling”).
- Para as densidades completas que não são conhecidas (e/ou difíceis de simular).

Algoritmo de metropolis hastings

- Defina uma distribuição (chamada de densidade proposta, de transição ou ainda “kernel”), digamos $q(\theta_j | \theta_j^{(r-1)})$, eg $\theta_j | \theta_j^{(r-1)} \sim N(\theta_j^{(r-1)}, \psi_\theta)$.
- Simule uma valor para θ_j a partir da densidade proposta, digamos $\theta_j^{(*)}$ e calcule

$$R_{\theta_j} = \frac{p(\theta_j^{(*)} | \theta_{-j}^{(r-1)}, \mathbf{x}) q(\theta_j^{(*)} | \theta_j^{(r-1)})}{p(\theta_j^{(r-1)} | \theta_{-j}^{(*)}, \mathbf{x}) q(\theta_j^{(r-1)} | \theta_j^{(*)})}$$

- Simule $u \sim U(0, 1)$, se $R_{\theta_j} \geq u$, faça $\theta_j^{(r)} = \theta_j^{(*)}$, caso contrário, faça $\theta_j^{(r)} = \theta_j^{(t-1)}$.

Voltando ao exemplo da regressão probito

- Considere a mesma estrutura anteriormente definida e

$$q(\beta|\beta^{(t-1)}) = N(\beta^{(t-1)}|\mathbf{\Sigma}_\beta)$$

- Note que a densidade é simétrica no sentido de que

$$q(\beta^{(*)}|\beta^{(t-1)}) = q(\beta^{(t-1)}|\beta^{(*)}).$$

- Assim, a taxa de aceitação dependerá apenas da distribuição condicional completa.
- Se a priori for proporcional à uma constante, a taxa de aceitação dependerá apenas da parte da verossimilhança que é função de β .

Algoritmo de MH no exemplo da regressão probito

- Simule $\beta^{(*)}$ a partir de $N(\beta^{(t-1)} | \Sigma_\beta)$ e calcule

$$R_\beta = \frac{\prod_{i=1}^n \Phi(\mathbf{X}'_i \beta^{(*)})^{y_i} (1 - \Phi(\mathbf{X}'_i \beta^{(*)}))^{1-y_i}}{\prod_{i=1}^n \Phi(\mathbf{X}'_i \beta^{(t-1)})^{y_i} (1 - \Phi(\mathbf{X}'_i \beta^{(t-1)}))^{1-y_i}} \\ \times \frac{\exp \left\{ -0,5 \left(\beta^{(*)} - \mu_\beta \right)' \Psi_\beta^{-1} \left(\beta^{(*)} - \mu_\beta \right) \right\}}{\exp \left\{ -0,5 \left(\beta^{(t-1)} - \mu_\beta \right)' \Psi_\beta^{-1} \left(\beta^{(t-1)} - \mu_\beta \right) \right\}}$$

- Simule $u \sim U(0, 1)$ e aceite $\beta^{(t)} = \beta^{(t-1)}$ se $R_\beta \geq u$, caso contrário, faça $\beta^{(t)} = \beta^{(*)}$.

Comentários

- Em geral, procura-se simular em blocos parâmetros que tendam a apresentar dependência à posteriori.
- Os valores para a variância da densidade proposta $q(.,.)$ devem ser escolhidos de tal forma que as taxas de aceitação não sejam nem muito elevadas nem muito pequenas.
- Melhor trabalhar com o logaritmo natural de R , e compará-lo com $\log(u)$, pois é bem mais estável do que trabalhar com o valor original.
- Existem variações do algoritmo de MH (no contexto de MCMC) que propõem formas alternativas de construção das densidades

propostas

Pacotes específicos

- O pacote MCMCpack possui a função MCMCprobit que permite o ajuste de modelos de regressão probito.
- Roda em *background* códigos feitos em C++ feitos a partir da biblioteca *Scythe Statistical Library*
- Segue comandos padrão do R.

Pacote geral

- Programa WinBugs: permite ajustar modelos complexos usando diversos algoritmos do tipo MCMC.
- Em geral, basta apenas fornecer o modelo (verossimilhança) e prioris.
- Limitações no uso de prioris impróprias (Jeffreys).
- Expedientes disponíveis para inserir verossimilhançar e prioris que não são padrão.
- Pode ser utilizado de modo mais simples através do pacote do R *R2WinBUGS*.

Algoritmos para simular das cond.(tradução do manual)

- Condicional completa contínua
 - Conjugada (conhecida): amostragem direta usando algoritmos padrão.
 - Log-côncava: Rejeição adaptativa de derivação livre (Gilks, 1992).
 - Espaço paramétrico restrito: Amostragem por corte (“slice-sampling”), Neal, 1997.
 - Espaço paramétrico restrito: Metropolis-Hastings (adaptativo).
- Condicional completa discreta
 - Limite superior finito: transformada inversa.
 - Poisson deslocada: amostragem direta usando algoritmos padrão.

Exemplo: regressão probito, dados aumentados

■ Código

Sem título

```
pbtmodel<-function(){  
  lo[1]<- -50  
  lo[2]<- 0  
  up[1]<- 0  
  up[2]<-50  
  for (i in 1 : N){  
    mu[i] <- beta0 + beta1 * x1[i] + beta2 * x2[i]+ beta3 * x3[i]  
    z[i] ~ dnorm(mu[i],1)%_I(lo[y[i]+1],up[y[i]+1])  
  }  
  beta0~dnorm(0,0.0001)  
  beta1~dnorm(0,0.0001)  
  beta2~dnorm(0,0.0001)  
  beta3~dnorm(0,0.0001)  
}
```


Exemplo: regressão probito, verossimilhança original

■ Código

Sem título

```
pbtmodel<-function(){  
  for (i in 1 : N){  
    y[i] ~ dbern(p[i])  
    logit(p[i]) <- beta0 + beta1 * x1[i] + beta2 * x2[i] +  
    beta3 * x3[i]  
  }  
  beta0~dnorm(0,0.0001)  
  beta1~dnorm(0,0.0001)  
  beta2~dnorm(0,0.0001)  
  beta3~dnorm(0,0.0001)  
}
```

Exemplo: regressão probito, verossimilhança original

truque da $P(X = 0)$, $X \sim \text{Poisson}(\phi)$

■ Código

Sem título

```
pbtmodel<-function(){  
  C<-10 # para garantir que a log-verossimilhança > 0  
  for (i in 1 : N){  
    zeros[i]<- 0  
    logit(p[i]) <- (beta0 + beta1 * x1[i] + beta2 * x2[i]+  
    beta3 * x3[i])  
    phi[i] <- -(y[i]*log(p[i])+(1-y[i])*log(1-p[i])) + C  
    zeros[i] ~ dpois(phi[i])  
  }  
  beta0~dnorm(0,0.0001)  
  beta1~dnorm(0,0.0001)  
  beta2~dnorm(0,0.0001)  
  beta3~dnorm(0,0.0001)  
}
```

Comandos para executar a simulação

- Comando geral para executar o algoritmo MCMC

```
simullBSgs = bugs(data, inits,
model.file="pbtmodel.bug",
parameters=c("beta0", "beta1", "beta2", "beta3"),
n.chains=1, n.iter=25000, n.burnin=1, n.thin=1,
bugs.directory="E:/windows/Programas_TRI/winBUGS14/",
codaPkg=FALSE)
```

Sem título

Comandos para executar a simulação

- `simullBSgs$sims.array[iteração,cadeia,parâmetro]`