

Relatório do Trabalho de Estatística Bayesiana I

Mistura de Normais com Variância Contaminada

Caio Balieiro
Taiguara Melo Tupinambás
Walmir dos Reis Miranda Filho

Prof. Dani Gamerman
Prof^a Rosangela Helena Loschi

Programa de Pós-Graduação em Estatística
Instituto de Ciências Exatas
Universidade Federal de Minas Gerais

Belo Horizonte, 2 de dezembro de 2019

1 Introdução

O presente trabalho tem como objetivo obter, dada uma densidade *a posteriori* conjunta dos parâmetros de um modelo probabilístico para uma amostra previamente observada, as densidades *a posteriori* marginais de cada parâmetro, bem como as estatísticas de média; variância; assimetria e curtose associadas, a partir da implementação de três métodos numéricos, a saber: (i) integração via quadratura de Riemann; (ii) reamostragem por importância sequencial (em inglês, *Sequential Importance Resampling*, ou SIR); e (iii) integração via Monte Carlo em cadeias de Markov (em inglês, *Markov Chain Monte Carlo*, ou MCMC) com inovações dadas pelo algoritmo de Metropolis-Hastings (MH).

Para o modelo que gerou a amostra observada, será pressuposto que sua especificação é dada por uma mistura finita de normais. Sejam X_1, \dots, X_n amostras aleatórias independentes, condicionalmente a um vetor de parâmetros $\theta = (\mu, \sigma^2, \nu)$, e identicamente distribuídas com função densidade dada por

$$f(x|\mu, \sigma^2, \nu) = \nu\phi(x|\mu, 100\sigma^2) + (1 - \nu)\phi(x|\mu, \sigma^2), \quad x \in \mathbb{R}, \quad (1)$$

onde $\phi(x|\mu, \sigma^2) = (2\pi\sigma^2)^{-1} \exp[-(x - \mu)^2/(2\sigma^2)]$ denota a função densidade da distribuição normal com média μ e variância σ^2 avaliada no ponto x . Para o suporte de cada parâmetro, tem-se que $\mu \in \mathbb{R}$, $\sigma^2 \in \mathbb{R}_+$ e $\nu \in (0, 1)$.

A mistura finita descrita em (1) possui duas componentes normais que diferem nas respectivas variâncias: para a primeira componente, o valor é 100 vezes o da segunda. Desta forma, tem-se uma mistura finita de normais com variância *contaminada*.

Para os parâmetros μ , σ^2 e ν , será pressuposto que cada um segue uma distribuição *a priori* predeterminada: $\mu|\sigma^2 \sim N(m, V\sigma^2)$, onde $N(\cdot)$ denota a distribuição normal com média $m \in \mathbb{R}$ e variância $V\sigma^2$, $V > 0$; $\sigma^2 \sim GI(a, d)$, onde $GI(\cdot)$ denota a distribuição gama inversa com parâmetros de forma $a > 0$ e de taxa $d > 0$ (inverso da escala); e $\nu \sim U(0, 1)$, a distribuição uniforme contínua padrão.

Para gerar uma amostra aleatória do modelo em (1), foi utilizada uma representação hierárquica (Lachos *et al.*, 2013)[1] tal que

$$X_i|\mu, \sigma^2, U_i = u_i \sim N(\mu, \sigma^2 u_i^{-1}), \quad U_i|\mu \sim p_d(1, 100) : P(U_i = 100) = \nu. \quad (2)$$

onde $p_d(a, b)$ denota uma função de probabilidade (discreta) que atribui massa probabilística apenas aos pontos a e b .

A partir do produto entre a função de verossimilhança $f(\mathbf{x}|\mu, \sigma^2, \nu)$ para uma amostra $\mathbf{x} = (x_1, \dots, x_n)$ gerada através da representação hierárquica em (2) e das densidades *a priori* para cada parâmetro do modelo, obtém-se o núcleo (em inglês, *kernel*) da densidade *a posteriori* conjunta $p(\mu, \sigma^2, \nu|\mathbf{x})$. Como a expressão da densidade *a posteriori* possui uma constante de proporcionalidade $f(x_1, \dots, x_n) = f(\mathbf{x})$ não facilmente calculável, mas que não depende dos parâmetros, pode-se usar o núcleo daquela para inferir sobre cada um dos parâmetros. No modelo de mistura finita de duas componentes normais com variância contaminada, temos que

$$\begin{aligned} p(\mu, \sigma^2, \nu|\mathbf{x}) &= \frac{f(\mathbf{x}|\mu, \sigma^2, \nu) \times p(\mu, \sigma^2, \nu)}{f(\mathbf{x})} \propto \prod_{i=1}^n f(x_i) \times p(\mu|\sigma^2) \times p(\sigma^2) \times p(\nu) \\ &\propto \prod_{i=1}^n [\nu\phi(x_i|\mu, 100\sigma^2) + (1 - \nu)\phi(x_i|\mu, \sigma^2)] \times \\ &\times \phi(\mu|m, V\sigma^2) \times \frac{d^a}{\Gamma(a)} \left(\frac{1}{\sigma^2}\right)^{a+1} \exp\left(-\frac{d}{\sigma^2}\right) \\ &\propto \left(\frac{1}{\sigma^2}\right)^{(n+1)/2+a+1} \exp\left\{-\frac{[(\mu - m)^2/(2V) + d]}{\sigma^2}\right\} \times A(\mathbf{x}|\mu, \sigma^2, \nu), \end{aligned} \quad (3)$$

onde

$$A(\mathbf{x}|\mu, \sigma^2, \nu) = \prod_{i=1}^n \left\{ \frac{\nu}{10} \exp \left[-\frac{(x_i - \mu)^2}{200\sigma^2} \right] + (1 - \nu) \exp \left[-\frac{(x_i - \mu)^2}{2\sigma^2} \right] \right\}$$

Evidentemente, trabalhar diretamente com o lado direito em (3) não é computacionalmente agradável, uma vez que esta é um produto de várias quantidades sempre menores do que 1, muitas delas próximas de 0, o que pode levar a problemas de aproximação numérica. Desta forma, sempre que for necessário calcular o núcleo da *posteriori*, dada uma amostra, inicialmente será tomada a soma do logaritmo de todas as quantidades que compõem este produto e só depois esta mesma soma será exponenciada. Dependendo do quão negativa for a soma (o logaritmo de termos entre 0 e 1 é sempre negativo), quando exponenciada ela pode ou não ser numericamente igual a 0.

Logo, na geração da amostra \mathbf{x} deve-se escolher valores de μ , σ^2 e ν tais que o núcleo de $p(\mu, \sigma^2, \nu|\mathbf{x})$ não seja numericamente nulo para os verdadeiros valores dos parâmetros. Considerando o uso da linguagem de programação *R* (R Core Team, 2019)[2], esta escolha não é difícil. Na versão 3.5.3, o menor valor que pode ser tomado como argumento da função `exp()` do *R* é -709 . Desta forma, basta tomar μ , σ^2 e ν tais que o logaritmo do lado direito em (3) seja maior do que -709 .

Para o presente trabalho, foram considerados uma amostra de tamanho $n = 500$ (Figura 1) da mistura finita de normais com variância contaminada parametrizada de tal forma que $\mu = 11$; $\sigma^2 = 0.64$ e $\nu = 0.2$, com hiperparâmetros $m = 11$; $V = 1$; $a = 7$ e $d = 4$ nas distribuições *a priori*. Fixada uma semente aleatória, para a amostra gerada a partir dos valores citados, o logaritmo do núcleo da densidade *a posteriori* foi igual a -518.9061 , cuja exponencial é aproximadamente igual a 4.385×10^{-226} .

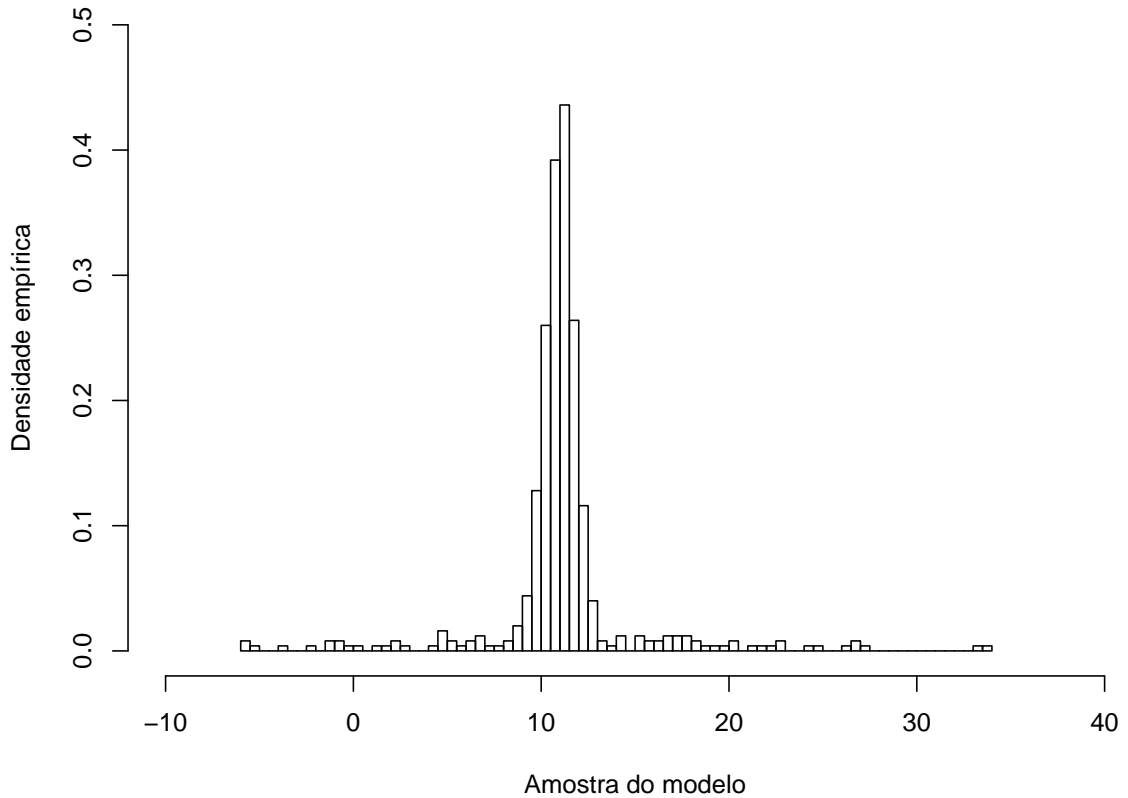
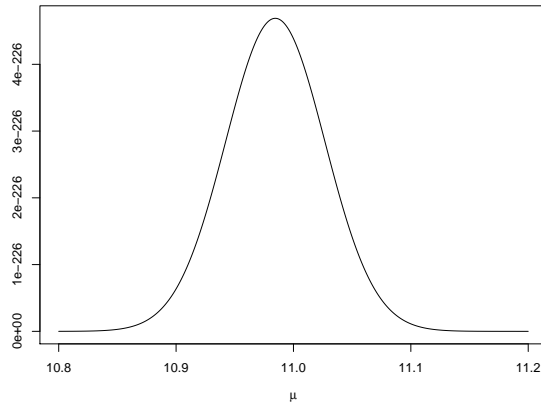


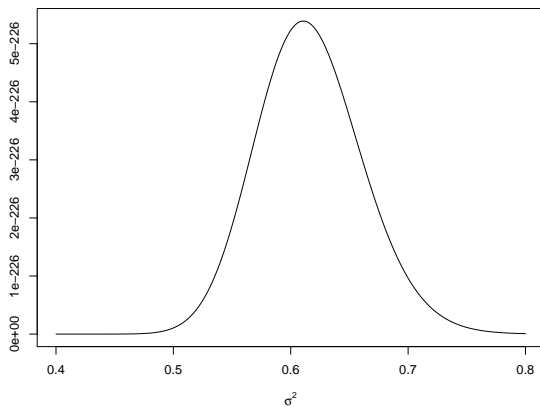
Figura 1: Histograma da amostra gerada do modelo

Como não se tem uma expressão fechada para $p(\mu, \sigma^2, \nu | \mathbf{x})$, mas apenas de seu núcleo, para obter as densidades *a posteriori* marginais de μ , σ^2 e ν dado \mathbf{x} , bem como as estatísticas associadas a cada uma delas, é necessário aproximá-las por algum método numérico. Na seção 2, é apresentado o primeiro dos três métodos considerados: a integração via quadratura de Riemann. Por sua vez, na seção 3 é descrito o método SIR. Por fim, na seção 4, é retratada a integração via MCMC com inovações MH. Todos os métodos serão testados e comparados no contexto do modelo exposto acima. Na seção 5, são resumidos os resultados obtidos para os três métodos, bem como é feita uma avaliação comparativa da qualidade de sua convergência.

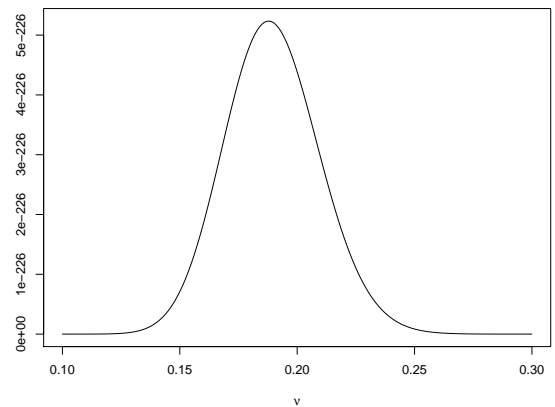
Antes de implementar cada um dos métodos numéricos, foram feitos três gráficos (Figuras 2a a 2c) do núcleo de $p(\mu, \sigma^2, \nu | \mathbf{x})$ quando se varia um dos três parâmetros e os outros dois são mantidos fixos nos respectivos valores verdadeiros pressupostos para o modelo que gerou a amostra. Tais gráficos permitem dizer, para cada parâmetro, quais são os intervalos do suporte correspondente que concentram praticamente toda a massa probabilística da densidade *a posteriori* marginal associada, ainda que as densidades calculadas sejam válidas apenas quando os outros dois parâmetros são fixados em valores arbitrários. Esta informação será útil para definir os intervalos de integração na quadratura de Riemann e a matriz de covariância da distribuição proposta nos métodos SIR e MCMC-MH para aproximação das densidades *a posteriori* marginais verdadeiras.



(a) $I_\mu = (10.85, 11.13)$, dados $\sigma^2 = 0.64, \nu = 0.2$



(b) $I_{\sigma^2} = (0.48, 0.78)$, dados $\mu = 11, \nu = 0.2$



(c) $I_\nu = (0.13, 0.26)$ dados $\mu = 11, \sigma^2 = 0.64$

Figura 2: Intervalos de massa probabilística para cada parâmetro (variável aleatória) do núcleo de $p(\mu, \sigma^2, \nu | \mathbf{x})$

2 O Método da Quadratura de Riemann

Em problemas multidimensionais, a densidade marginal correspondente a cada variável aleatória para a qual se deseja fazer inferência é obtida integrando a densidade conjunta sobre as demais. Frequentemente, o cálculo das integrais pode ser difícil ou mesmo impossível analiticamente. No caso de uma densidade *a posteriori* conjunta deve-se aproximar algumas das, ou todas as, integrais associadas a cada densidade *a posteriori* marginal de interesse por métodos numéricos, os quais podem ser determinísticos ou estocásticos. Neste presente trabalho, isto será feito para todos os 3 parâmetros do modelo de mistura finita de normais com variância contaminada.

Dentre os métodos determinísticos mais usuais, para a aproximação das integrais correspondentes às densidades *a posteriori* marginais de μ , σ^2 e ν , será utilizado o método da quadratura de Riemann, um caso particular e o mais simples da família das *fórmulas de Newton-Cotes* (Chapra, 2015)[3], as quais substituem o verdadeiro integrando por uma aproximação polinomial dentro de cada subintervalo contido no intervalo de integração. Na quadratura de Riemann, o integrando avaliado em cada subintervalo é aproximado por uma função constante avaliada no limite inferior (ou superior) do mesmo subintervalo.

Aproximações polinomiais com nós igualmente espaçados a cada subintervalo, como as regras de quadratura do Trapézio (grau 1) ou de Simpson (grau 2), obtém resultados com menor erro do que a quadratura de Riemann. Quando o grau da aproximação polinomial tende ao infinito, a quadratura converge para o verdadeiro valor da integral. Porém, em problemas de dimensão igual a 2 ou superior, usar aproximações polinomiais com nós torna-se muito custoso computacionalmente, uma vez que temos três ou mais termos na soma que aproximará a integral para cada dimensão. Por esta razão, escolheu-se a quadratura de Riemann para obter densidades *a posteriori* marginais de μ , σ^2 e ν , pois esta regra possui um único termo na soma aproximadora, facilitando a programação das rotinas iterativas associadas. Em uma dada aproximação, o mesmo número de intervalos será usado para todas as dimensões.

Antes de aproximar as densidades *a posteriori* marginais de cada parâmetro, é necessário aproximar o inverso da constante de proporcionalidade. Dados três parâmetros $(\alpha_1, \alpha_2, \alpha_3)$ e uma amostra dos dados \mathbf{y} quaisquer, suponha que se deseja aproximar a densidade *a posteriori* marginal de α_3 dados os pontos r_i, s_j, t_k da grade formada por todos os subintervalos de integração, $i, j, k \in \{1, \dots, L\}$. Temos pela quadratura de Riemann que

$$\begin{aligned} p(\alpha_3|\mathbf{y}) &= \iint p(\alpha_1, \alpha_2, \alpha_3|\mathbf{y}) d\alpha_1 d\alpha_2 \\ \Rightarrow p(t_k|\mathbf{y}) &= \iint p(\alpha_1, \alpha_2, t_k|\mathbf{y}) d\alpha_1 d\alpha_2 \approx \sum_{i=1}^L \sum_{j=1}^L p(r_i, s_j, t_k|\mathbf{y}) \Delta_i \Delta_j \\ &= \sum_{i=1}^L \sum_{j=1}^L c \cdot h(r_i, s_j, t_k|\mathbf{y}) \Delta_i \Delta_j \end{aligned} \quad (4)$$

Como o inverso de c , a constante de proporcionalidade, é dado pela densidade *a priori* preditiva $f(\mathbf{y})$, a qual é obtida integrando-se o núcleo resultante do produto entre a função de verossimilhança $f(\mathbf{y}|\alpha_1, \alpha_2, \alpha_3)$ e as densidades (ou funções de probabilidade) *a priori* para α_1 , α_2 e α_3 , também é possível aproximar c^{-1} pela quadratura de Riemann. Neste caso, $c^{-1} = \sum_{i=1}^L \sum_{j=1}^L \sum_{k=1}^L h(r_i, s_j, t_k|\mathbf{y}) \Delta_i \Delta_j \Delta_k$. Com o valor aproximado para c , é possível calcular (4) nos limites superior e inferior de todos os subintervalos de um dado parâmetro e enfim obter uma aproximação da densidade *a posteriori* marginal deste mesmo parâmetro através de uma curva gráfica que liga todos os valores calculados. Evidentemente, quanto menor o tamanho de cada subintervalo, melhor a curva traçada aproximará a verdadeira densidade.

Para o modelo de mistura finita de normais com variância contaminada,

- 3 O Método da Reamostragem Por Importância Sequencial (SIR)
- 4 O Método de Monte Carlo via Cadeias de Markov (MCMC)
- 5 Considerações Finais

Referências

- [1] Lachos, V. H.; Castro, L. M. & Dey, D. K. Bayesian inference in nonlinear mixed-effects models using normal independent distributions. *Computational Statistics and Data Analysis*, 64(16):237–252, 2013.
- [2] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2019.
- [3] Chapra, S. C. & Canale, R. P. *Numerical Methods for Engineers*. McGraw-Hill Education, New York, NY, 7th edition, 2015.

Apêndice: Código R para os Resultados Apresentados

Introdução

```
# Tamanho amostral "n"; parâmetros "mu", "s2" ( $\sigma^2$ ) e "nu" do modelo para  
# a distribuição amostral e hiperparâmetros m, V, a e d das distribuições "a  
# priori" (somente para "mu" e "s2"):
```

```
n = 500; mu = 11; s2 = 0.64; nu = 0.2; m = 11; V = 1; a = 7; d = 4
```

```
# Geração hierárquica da amostra de tamanho "n" do modelo:
```

```
U_pdf = function(n, nu) {  
  u = sample(c(100, 1), prob=c(nu, 1-nu), size=n, replace=T)  
}
```

```
set.seed(122019)
```

```
u = U_pdf(n, nu)  
sam = rnorm(n, mu, sqrt(s2*u))
```

```
# Histograma da amostra gerada do modelo (figura 1):
```

```
par(mar = c(5,5,3,2))  
hist(sam, breaks=100, prob=T, main="", xlim = c(-10, 40), ylim = c(0, 0.5),  
     xlab = "Amostra do modelo", ylab = "Densidade empírica")
```

```
# Cálculo do logaritmo do núcleo da distribuição "a posteriori" original:
```

```
logA = function(X, mu, sigma2, nu) {  
  n = length(X)  
  k1 = (nu/10)*exp(-(X-mu)^2/(2*100*sigma2))  
  k2 = (1-nu)*exp(-(X-mu)^2/(2*sigma2))  
  k = log(k1 + k2)  
  return(k)  
}
```

```
logh = function(n, mu, sigma2, nu, m, V, a, d) {  
  k1 = -((n + 1)/2 + a + 1)*log(sigma2)  
  k2 = -((mu - m)^2/(2*V) + d)/sigma2  
  k = k1 + k2  
  return(k)  
}
```

```
logkpost = function(X, mu, sigma2, nu, m, V, a, d) {  
  n = length(X)  
  lA = logA(X, mu, sigma2, nu)  
  lh = logh(n, mu, sigma2, nu, m, V, a, d)  
  lkp = sum(lA) + lh  
  return(lkp)  
}
```

```
# Gráficos dos intervalos de massa probabilística para cada parâmetro do
# núcleo da distribuição "a posteriori", fixados os demais, após algumas
# tentativas anteriores para redução dos limites do gráfico e melhor vi-
# sualização da curva (figuras 2a a 2c):
```

```
mu_sup=seq(10.8, 11.2, 0.001); t_mu=length(mu_sup); kp_mu=numeric(t_mu)
```

```
for(i in 1:t_mu) {
kp_mu[i] = exp(logkpost(X=sam,
mu=mu_sup[i], sigma2=s2, nu=nu, m=m, V=V, a=a, d=d))
}
plot(mu_sup, kp_mu, type="l", main="",
      xlab=expression(paste(mu)), ylab="")
```

```
s2_sup=seq(0.4, 0.8, 0.001); t_s2=length(s2_sup); kp_s2=numeric(t_s2)
```

```
for(i in 1:t_s2) {
kp_s2[i] = exp(logkpost(X=sam,
mu=mu, sigma2=s2_sup[i], nu=nu, m=m, V=V, a=a, d=d))
}
plot(s2_sup, kp_s2, type="l", main="",
      xlab=expression(paste(sigma^2)), ylab="")
```

```
nu_sup=seq(0.1, 0.3, 0.001); t_nu=length(nu_sup); kp_nu=numeric(t_nu)
```

```
for(i in 1:t_nu) {
kp_nu[i] = exp(logkpost(X=sam,
mu=mu, sigma2=s2, nu=nu_sup[i], m=m, V=V, a=a, d=d))
}
plot(nu_sup, kp_nu, type="l", main="",
      xlab=expression(paste(nu)), ylab="")
```

O Método da Quadratura de Riemann

```
# Cenários considerados, variando no número de subintervalos utilizados:
```

```
L1 = 15; L2 = 50; L3 = 100
```

```
# Tamanhos dos subintervalos, dada a combinação de cenário e parâmetro, e
# grades definidas pelos mesmos:
```

```
mu_step1 = (11.13 - 10.85)/L1; mu_gr1 = seq(10.85, 11.13, mu_step1)
mu_step2 = (11.13 - 10.85)/L2; mu_gr2 = seq(10.85, 11.13, mu_step2)
mu_step3 = (11.13 - 10.85)/L3; mu_gr3 = seq(10.85, 11.13, mu_step3)
```

```
s2_step1 = (0.78 - 0.48)/L1; s2_gr1 = seq(0.48, 0.78, s2_step1)
s2_step2 = (0.78 - 0.48)/L2; s2_gr2 = seq(0.48, 0.78, s2_step2)
s2_step3 = (0.78 - 0.48)/L3; s2_gr3 = seq(0.48, 0.78, s2_step3)
```



```

nu_step1 = (0.26 - 0.13)/L1; nu_gr1 = seq(0.13, 0.26, nu_step1)
nu_step2 = (0.26 - 0.13)/L2; nu_gr2 = seq(0.13, 0.26, nu_step2)
nu_step3 = (0.26 - 0.13)/L3; nu_gr3 = seq(0.13, 0.26, nu_step3)

# Combinando as três grades unidimensionais em cada cenário, todas com
# número igual de subintervalos e de pontos (nos limites dos interval-
# os) para integração numérica:

grid_tri1 = cbind(mu_gr1, s2_gr1, nu_gr1); l1 = nrow(grid_tri1)
grid_tri2 = cbind(mu_gr2, s2_gr2, nu_gr2); l2 = nrow(grid_tri2)
grid_tri3 = cbind(mu_gr3, s2_gr3, nu_gr3); l3 = nrow(grid_tri3)

# Produto triplo dos tamanhos dos subintervalos em cada cenário:

prod_step1 = mu_step1*s2_step1*nu_step1
prod_step2 = mu_step2*s2_step2*nu_step2
prod_step3 = mu_step3*s2_step3*nu_step3

# Cálculo do inverso da constante de proporcionalidade em cada cenário:

cprop = function(l, X, mgr, s2gr, ngr, prst, m, V, a, d){
  c = 0
  for (i in 1:l) {
    for (j in 1:l) {
      for (k in 1:l) {
        aux = logkpost(X, mgr[i], s2gr[j], ngr[k], m, V, a, d)
        c = c + exp(aux)*prst
      }
    }
  }
  return(c)
}

c1 = cprop(l=l1, X=sam, mgr=mu_gr1, s2gr=s2_gr1, ngr=nu_gr1,
  prst=prod_step1, m=m, V=V, a=a, d=d)
c2 = cprop(l=l2, X=sam, mgr=mu_gr2, s2gr=s2_gr2, ngr=nu_gr2,
  prst=prod_step2, m=m, V=V, a=a, d=d)
c3 = cprop(l=l3, X=sam, mgr=mu_gr3, s2gr=s2_gr3, ngr=nu_gr3,
  prst=prod_step3, m=m, V=V, a=a, d=d)
c1; c2; c3

```

O Método da Reamostragem Por Importância Sequencial (SIR)

a

O Método de Monte Carlo via Cadeias de Markov (MCMC)

a