



UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE MATEMÁTICA
DEPARTAMENTO DE ESTATÍSTICA



Aproximações de distribuições marginais a posteriori utilizando o método INLA.

Autor: Douglas Roberto Mesquita Azevedo
Orientador: Professora Dra. Vanessa Bielefeldt Leotti Torman.

Porto Alegre, 20 de Dezembro de 2013.

Universidade Federal do Rio Grande do Sul
Instituto de Matemática
Departamento de Estatística

Aproximações de distribuições marginais a posteriori
utilizando o método INLA.

Douglas Roberto Mesquita Azevedo

Monografia apresentada para obtenção do grau
de Bacharel em Estatística

Banca examinadora:

Professora Dra. Vanessa Bielefeldt Leotti Torman.

Professora Dra. Patricia Klarmann Ziegelmann.

Porto Alegre, 20 de Dezembro de 2013

Dedico este trabalho aos meus pais, Ivani e Paulo, e aos meus irmãos, Gretchem e Julian.

Agradecimentos

Agradeço primeiramente aos meus pais, Ivani e Paulo, pela confiança, por estarem presentes em todos os passos da minha trajetória e pelo apoio incondicional em todas as minhas escolhas.

Agradeço aos meus irmãos, Gretchem e Julian, que mesmo distantes geograficamente, sempre se mantiveram presentes me incentivando, motivando e ajudando sempre que necessário.

Agradeço aos meus amigos, por compreenderem minha ausência, pelo acolhimento quando pude estar presente e pelos bons momentos proporcionados. Em especial aos amigos Douglas Vargas, Gulliti Sena e Mateus Essvein.

Agradeço em especial ao amigo, colega e sócio Tiago Lenhard, pela grande amizade desenvolvida nos últimos anos, por sempre estar disponível para tudo e por ter estado presente nas minhas últimas grandes decisões, sempre me incentivando e me ajudando a ver o lado bom de tudo.

Agradeço aos colegas de curso pelos momentos de descontração e por fazerem parte da minha história. Em especial a Bárbara Pederiva pela amizade dentro e fora da universidade e aos amigos Allan Lemos, Felipe Costa, Jorge Staudt, Lucas Freitas e Suziane Pereira pelas muitas risadas proporcionadas.

Agradeço aos professores que pude conhecer durante estes quatro anos de curso. Em especial ao professor Marcio Valk, por ter me acolhido como bolsista e ter me apresentado a área da pesquisa científica, a professora Patricia Ziegelmann por diversas vezes ter me ajudado ao longo do curso e por ter aceito ser avaliadora deste trabalho e a Professora Vanessa Tormann por servir de inspiração como excelente profissional e por ter sido minha orientadora neste trabalho.

Agradeço ao professor Nico pelos ensinamentos durante o ensino médio e por me motivar a seguir na área das ciências exatas.

Agradeço a todas as outras pessoas que de uma forma ou de outra contribuíram para o meu crescimento e me apoiaram nesta jornada. Em especial ao meu cunhado Louis por ter me ajudado sempre que necessário.

Resumo

O uso da Estatística Bayesiana vem se tornando cada vez mais frequente, isso pois os modelos são mais intuitivos e em alguns casos apenas modelos Bayesianos obtêm sucesso (modelos de Teoria da Resposta ao Item por exemplo). O grande problema surge quando não se pode derivar analiticamente os resultados, o que ocorre para a maioria dos modelos. Para driblar esta barreira, os métodos mais difundidos foram os métodos de simulação MCMC (sigla em inglês de *Markov Chain Monte Carlo*, que significa Monte Carlo via Cadeias de Markov). O grande contraponto destes métodos é o tempo computacional demandado, que por vezes torna a análise de modelos mais complexos demasiadamente demorada.

Neste sentido o método INLA, que é um método determinístico, surge como uma solução para os problemas pois, quando comparado com os métodos de simulação obtêm resultados muito satisfatórios e com um tempo computacional mais viável, o que facilita a prática da Estatística Bayesiana.

O principal objetivo deste trabalho é iniciar o leitor no método tanto nos princípios básicos teóricos quando na parte computacional. No presente trabalho utilizamos alguns exemplos para ilustrar os resultados matemáticos em que o método INLA se baseia. Além disto, apresenta-se um breve tutorial com alguns comandos para executar o método já implementado no pacote *R-INLA* do software R.

Durante este trabalho foi possível observar que a matemática inserida no método é complexa e que os resultados são bastante satisfatórios, tanto quando comparados com resultados exatos (quando disponíveis) como quando comparados com resultados via MCMC. Também foi possível perceber que a utilização do pacote *R-INLA* é bastante simples e o tempo computacional gasto para geração dos resultados é mais baixo do que o tempo computacional gasto em métodos de simulação.

Sumário

1	Introdução e conceitos básicos	7
1.1	Teorema de Bayes	8
1.2	Inferência Bayesiana	8
1.3	Método de Laplace	10
1.4	Aproximação Gaussiana	13
2	Introdução ao método INLA	17
2.1	Aproximação para $\pi(\theta y)$	19
2.2	Aproximação para $\pi(x_i \theta, y)$	19
2.3	<i>R-INLA</i>	20
2.4	Exemplos	21
2.4.1	Modelo Binomial	22
2.4.2	Modelo Normal	31
3	Aplicações	39
3.1	Regressão Logística com efeitos aleatórios	39
3.1.1	Comparação com o WinBUGS.	47
3.2	Componentes de Variância	49
3.2.1	Comparação com o WinBUGS	53
3.3	Medidas Repetidas	54
3.3.1	Comparação com o WinBUGS	61
4	Conclusões	63
5	Referências	64
6	Anexos	65

Lista de Figuras

1	Comparação entre o Método de Laplace e o Método INLA para o modelo Binomial.	13
2	Aproximação gaussiana para a distribuição a posteriori do parâmetro p de uma Binomial.	16
3	Distribuição a posteriori para o logito da probabilidade de sucesso no modelo Binomial utilizando uma grade contendo 1000 pontos.	26
4	Distribuição a posteriori do parâmetro η para algumas combinações de k , n e verdadeira proporção (p).	28
5	Comparação entre posterioris para o parâmetro p , utilizando priori Normal e Beta não informativas	29
6	Distribuição a posteriori do parâmetro p para algumas combinações de k , n e verdadeira proporção (p).	30
7	Distribuição a posteriori para o logaritmo da precisão do modelo Normal utilizando uma grade contendo 100 pontos.	36
8	Distribuição a posteriori normalizada para a média do modelo Normal utilizando uma grade contendo 100 pontos.	37
9	Distribuição a posteriori do parâmetro Média para algumas combinações de verdadeira Média, Variância e tamanho de amostra.	38
10	Distribuição marginal a posteriori para o logaritmo da probabilidade de morte de bebês em cirurgias cardíacas.	43
11	Distribuição marginal a posteriori para a probabilidade de morte de bebês em cirurgias cardíacas.	44
12	Efeitos aleatórios para os 12 hospitais estudados. Pontos indicam os quantis (0.025, 0.5 e 0.975) e o traço indica a média.	45
13	Distribuição marginal a posteriori para a precisão dos efeitos aleatórios do modelo de regressão logística com efeitos aleatórios.	46
14	Distribuição marginal a posteriori para a variância dos efeitos aleatórios do modelo de regressão logística com efeitos aleatórios.	47
15	Distribuição marginal a posteriori para a precisão dos efeitos aleatórios do modelo de componentes de variância.	52
16	Distribuição marginal a posteriori para a variância dos efeitos aleatórios do modelo de componentes de variância.	53
17	Distribuição marginal a posteriori para a precisão dos efeitos aleatórios e para correlação do modelo de medidas repetidas.	58

18	Distribuição a posteriori marginal para a variância dos efeitos aleatórios do modelo de medidas repetidas.	59
19	Distribuição marginal a posteriori par os efeitos fixos do modelo de medidas repetidas.	60
20	Comparação dos efeitos aleatórios do Modelo de Medidas Repetidas estimados por INLA e WinBUGS.	62

1 Introdução e conceitos básicos

Atualmente, os métodos de Estatística Bayesiana estão ganhando uma maior visibilidade pelos pesquisadores, devido a sua simples estrutura e a sua alta aplicabilidade nas mais diversas áreas do conhecimento, além de possuírem interpretações mais intuitivas do que as da Estatística Clássica. Sua aplicação no entanto exige um bom suporte computacional, pois os cálculos algébricos em geral são inviáveis.

A maioria dos modelos ajustados utilizando Estatística Bayesiana são analiticamente impossíveis de serem resolvidos. Para driblar essa barreira muitos recursos numéricos e computacionais foram apresentados. Como possíveis soluções podemos citar métodos determinísticos, como por exemplo o método de Laplace proposto por Tierney e Kadane [10] onde todas as integrais são calculadas através de um método numérico. Ou então podemos citar o uso de métodos de simulação, como por exemplo o método *Gibbs Sampling* [2], onde integrais não se fazem necessárias.

Os métodos que mais se difundiram foram os métodos de simulação chamados de MCMC (*Markov Chain Monte Carlo*), dentre os quais o já citado *Gibbs Sampling*. A ideia destes métodos é utilizar cadeias de Markov para se obter uma distribuição limite para os parâmetros do modelo. Para isso se constrói um processo iterativo visando coletar amostras aleatórias das distribuições de interesse sem a necessidade de resolver algebricamente todos os cálculos. Estes métodos conseguem obter resultados satisfatórios, porém com o contraponto de em determinados casos precisarem de longas cadeias para obter convergência, o que torna o processo demorado. Para mais informações sobre métodos MCMC ver [5].

No ano de 2009 um novo método determinístico foi proposto, o método INLA (*Integrated Nested Laplace Approximations*) [9]. Este método é aplicável numa grande gama de modelos, chamados de Modelos Latentes Gaussianos. O desempenho do método e o tempo computacional são satisfatórios, tendo resultados similares aos métodos de simulação de forma muito mais rápida, além de não ser necessário a verificação de convergência.

Este trabalho tem por objetivo ilustrar de maneira acessível, através de exemplos, a matemática inserida no método INLA, e ao mesmo tempo, servir como um breve tutorial sobre os comandos deste método utilizando o pacote *R-INLA* do software estatístico R.

Num primeiro momento falaremos sobre assuntos básicos para a compreensão das ideias abordadas durante o trabalho, na sequência será introduzido o método INLA bem como a notação a ser utilizada durante o trabalho. As duas últimas seções irão mostrar através de exemplos a matemática inserida no método INLA e a programação necessária para implementação de alguns exemplos práticos.

1.1 Teorema de Bayes

O Teorema de Bayes é útil quando estamos utilizando probabilidades condicionais.

Teorema de Bayes [6]: Considere o espaço de probabilidade $(\Omega, \mathcal{A}, P(\cdot))$. Seja B_1, B_2, \dots, B_n uma partição de Ω . Isto é, $(B_i \cap B_j = \emptyset$ para todo e qualquer $i \neq j)$ e $P(\bigcup_{i=1}^{\infty} B_i) = 1$. Então, para todo $A \in \mathcal{A}$ no qual $P(A) > 0$:

$$P[B_k|A] = \frac{P[B_k \cap A]}{P[A]} = \frac{P[A|B_k] P[B_k]}{\sum_{j=1}^n P[A|B_j] P[B_j]}$$

Este teorema mostra que é possível atualizar a informação de um determinado evento, modificando assim sua probabilidade de ocorrência. Seja B_k um evento de interesse, digamos que um determinado evento A já tenha ocorrido, podemos utilizar esta informação para atualizar a probabilidade de B_k através de probabilidades condicionais, caso A e B_k sejam eventos independentes então $P(B_k|A) = P(B_k)$, ou seja se a ocorrência de A não influencia na ocorrência de B_k , então o evento A não está trazendo nenhuma informação relevante para a probabilidade de ocorrência de B_k .

Este teorema é a base do processo de inferência em Estatística Bayesiana, e é utilizado para combinar as informações provenientes da amostra com a informação prévia a respeito de parâmetros desconhecidos dos modelos, como será visto na próxima subseção.

1.2 Inferência Bayesiana

Podemos dividir a Estatística em duas grandes correntes, onde a maior diferença entre elas é dada pela forma de considerarmos os parâmetros dos modelos:

- **Estatística Clássica:** Consideramos fixos os parâmetros desconhecidos do modelo.
- **Estatística Bayesiana:** Consideramos aleatórios os parâmetros desconhecidos do modelo.

Em Estatística Clássica contamos apenas com a informação contida na amostra para encontrarmos estimativas para os parâmetros, já em Estatística Bayesiana podemos acrescentar informações prévias a respeito dos parâmetros através de distribuições de probabilidade. Tal informação prévia é utilizada juntamente com a informação contida na amostra e com o auxílio do Teorema de Bayes podemos combinar as informações e obter a distribuição a posteriori dos parâmetros,

$$\pi(\theta|y) = \frac{\pi(\theta)L(y;\theta)}{\int_{\Theta} \pi(\theta)L(y;\theta)d\theta} \quad (1)$$

onde θ é o parâmetro desconhecido do modelo (podendo ser um escalar ou um vetor), $\pi(\theta)$ é a distribuição a priori para θ , $L(y; \theta)$ é a função de verossimilhança de θ e $\pi(\theta|y)$ a distribuição a posteriori para θ . Vejamos mais detalhes:

- **Distribuição a Priori**

Toda informação prévia a respeito dos parâmetros (θ) pode ser inserida no modelo através de uma distribuição de probabilidades. Essa distribuição é chamada de distribuição a Priori e é denotada por $\pi(\theta)$. Nesta distribuição deve estar contida toda informação que se tenha sobre os parâmetros, sempre estando atento para atribuir uma priori que contemple o intervalo de possíveis valores assumidos pelos parâmetros. Caso não se tenha nenhuma informação prévia, pode-se ainda utilizar uma priori que dê a mesma probabilidade para todos os possíveis valores do parâmetro, neste caso teremos então uma priori não informativa.

- **Função de Verossimilhança**

A função de Verossimilhança é uma função dos parâmetros do modelo e agrega toda a informação contida na amostra, caso tenhamos observações independentes e identicamente distribuídas da uma determinada distribuição $f(y; \theta)$, onde Y é um vetor de observações (Y_1, Y_2, \dots, Y_n) a função de verossimilhança será dada por,

$$L(y; \theta) = \prod_{i=1}^n f(y_i; \theta) = f(y; \theta)^n$$

- **Distribuição a Posteriori**

Essa é a distribuição de maior interesse em Inferência Bayesiana. É uma combinação da distribuição a priori com a função de verossimilhança dos dados, e descreve a incerteza a respeito do parâmetro θ após ser observada a amostra. Quando conseguimos obter esta distribuição, é possível se obter estimativas pontuais a posteriori para o parâmetro do modelo tais como a média, a mediana e a moda, também é possível calcular intervalos de credibilidade que podem ser de dois tipos:

- **Central:** Fixa-se um valor de probabilidade, em seguida encontram-se os limites do intervalo de forma que em ambas as caudas da distribuição a posteriori tenhamos uma área com metade da probabilidade escolhida.
- **HPD (do inglês *Highest Probability Density*, que significa maior densidade de probabilidade):** Fixa-se um valor de probabilidade e escolhe-se o intervalo de menor amplitude cuja área entre os limites seja igual a probabilidade escolhida.

Na Equação (1) o denominador é constante em relação a θ , então, podemos dizer que $\pi(\theta|y) \propto \pi(\theta)L(y;\theta)$. Isto é, os dois lados da equação tem o mesmo formato, diferindo apenas em escala.

Frequentemente nossos modelos possuem mais de um parâmetro, desta forma, ao utilizarmos a metodologia acima chegamos na distribuição a posteriori conjunta para o vetor de parâmetros θ . No entanto, nosso interesse está em conhecer a distribuição a posteriori marginal de cada um destes parâmetros, definida por:

$$\pi(\theta_i|y) = \int \pi(\theta|y) d\theta_{-i} \quad (2)$$

onde θ_{-i} se refere a todos os parâmetros exceto o parâmetro i em questão.

Em teoria esses seriam os passos a se seguir para se obter as distribuições marginais a posteriori para os parâmetros. Porém, na maioria dos casos não é possível efetuar a integração analítica do denominador da Equação (1) e então outras soluções devem ser avaliadas.

A seguir uma das aproximações determinísticas para distribuições a posteriori, o Método de Laplace, será apresentado.

1.3 Método de Laplace

O método de Laplace é um método determinístico para obtenção de distribuições marginais a posteriori. Foi proposto por Tierney e Kadane em 1986 [10]. Este método aproxima numericamente integrais da seguinte forma:

$$\int_x \exp\{f(x)\} dx \approx \sqrt{\frac{2\pi}{|f''(x_0)|}} \exp\{f(x_0)\}$$

onde x_0 é o ponto de máximo da função $f(x)$ e $f''(x_0)$ é a segunda derivada da função $f(x)$ aplicada num ponto x_0 , sendo que o melhor ponto para se aplicar é a moda da distribuição.

Para exemplificação será utilizado o modelo Binomial. Seja Y uma variável aleatória com distribuição Binomial com n fixo e conhecido e probabilidade de sucesso p . Seja η o logito da probabilidade de sucesso, isto é, $\eta = \ln\left(\frac{p}{1-p}\right)$. Desta forma o parâmetro p pode ser reescrito como $\left(\frac{\exp(\eta)}{1+\exp(\eta)}\right)$. A função de verossimilhança de k observações dessa variável aleatória é:

$$L(y; \eta) = \prod_{i=1}^k C_{y_i}^n \frac{\exp(\eta)^{y_i}}{(1 + \exp(\eta))^{y_i}} \left(1 - \frac{\exp(\eta)}{(1 + \exp(\eta))}\right)^{(n-y_i)} = \left(\prod_{i=1}^k C_{y_i}^n\right) \exp(\eta) \sum_{i=1}^k y_i (1 + \exp(\eta))^{-nk}$$

Como $0 \leq p \leq 1$, o logito dessa probabilidade pode assumir qualquer valor real. Assim, como priori para η pode ser utilizada a distribuição Normal(μ, σ^2) cuja função densidade de probabilidade é dada por:

$$\pi(\eta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(\eta - \mu)^2\right\}$$

Com estas duas equações e utilizando o teorema de Bayes podemos encontrar a distribuição a posteriori conjunta para η .

$$\pi(\eta|y) = \frac{\pi(\eta)L(y;\eta)}{\int_{-\infty}^{\infty} \pi(\eta)L(y;\eta)d\eta}$$

Substituindo as equações e realizando algumas simplificações obtemos:

$$\pi(\eta|y) = \frac{\exp(\eta)\sum_{i=1}^k y_i (1 + \exp(\eta))^{-nk} \exp\left\{-\frac{1}{2\sigma^2}(\eta - \mu)^2\right\}}{\int_{-\infty}^{\infty} \exp(\eta)\sum_{i=1}^k y_i (1 + \exp(\eta))^{-nk} \exp\left\{-\frac{1}{2\sigma^2}(\eta - \mu)^2\right\} d\eta}$$

Primeiramente é necessário resolver a integral contida no denominador, para assim obter a posteriori de η . Posteriormente pode-se transformar essa posteriori para encontrar a distribuição para p .

$$\int_{-\infty}^{\infty} \exp(\eta)\sum_{i=1}^k y_i (1 + \exp(\eta))^{-nk} \exp\left\{-\frac{1}{2\sigma^2}(\eta - \mu)^2\right\} d\eta \quad (3)$$

No entanto, tal integral não pode ser resolvida analiticamente. Para obtermos esta integral utilizando o método de Laplace precisamos aplicar a função logarítmica na função do integrando da Equação 3, a qual chamaremos de $f(\eta)$.

$$f(\eta) = \log\left(\exp(\eta)\sum_{i=1}^k y_i (1 + \exp(\eta))^{-nk} \exp\left\{-\frac{1}{2\sigma^2}(\eta - \mu)^2\right\}\right)$$

Como descrito no método, é necessário aplicar a aproximação numérica num ponto η_0 . Encontramos o ponto η_0 derivando a função $f(\eta)$ em relação a η e igualando a 0 (ponto de máximo).

$$f'(\eta) = \sum_{i=1}^k y_i - nk \frac{\exp(\eta)}{(1 + \exp(\eta))} - \frac{1}{\sigma^2}(\eta - \mu)$$

Não é possível obter o ponto de máximo desta função analiticamente, porém vamos assumir que o ponto de máximo η_0 possa ser encontrado através de algum método de otimização numérico, como Newton-Raphson por exemplo.

Precisamos também da segunda derivada da $f(\eta)$.

$$f''(\eta) = -nk \frac{\exp(\eta)}{(1 + \exp(\eta))^2} - \frac{1}{\sigma^2}.$$

Vamos também definir $\gamma_0 = \frac{1}{\sqrt{|f''(x)|}}$,

$$\gamma_0 = \frac{1}{\sqrt{\left| \left(-nk \frac{\exp(\eta_0)}{(1+\exp(\eta_0))^2} - \frac{1}{\sigma^2} \right) \right|}}$$

Desta forma,

$$\int_{-\infty}^{\infty} \exp(\eta)^{\sum_{i=1}^k y_i} (1 + \exp(\eta))^{-nk} \exp \left\{ -\frac{1}{2\sigma^2} (\eta - \mu)^2 \right\} d\eta \approx \sqrt{2\pi} \gamma_0 \exp \left\{ \eta_0 \sum_{i=1}^k y_i - nk \ln (1 + \exp(\eta_0)) - \frac{1}{2\sigma^2} (\eta_0 - \mu)^2 \right\}$$

Resolvida a integral do denominador temos então a seguinte distribuição a posteriori para η .

$$\pi(\eta|y) \approx \frac{\exp \left\{ \eta \sum_{i=1}^k y_i - nk \ln (1 + \exp(\eta)) - \frac{1}{2\sigma^2} (\eta - \mu)^2 \right\}}{\sqrt{2\pi} \gamma_0 \exp \left\{ \eta_0 \sum_{i=1}^k y_i - nk \ln (1 + \exp(\eta_0)) - \frac{1}{2\sigma^2} (\eta_0 - \mu)^2 \right\}}$$

A distribuição de p pode ser obtida pelo método Jacobiano, já que $p = \frac{\exp(\eta)}{(1+\exp(\eta))}$.

$$\pi(p|y) \approx \frac{\exp \left\{ \ln \left(\frac{p}{1-p} \right) \sum_{i=1}^k y_i - nk \ln \left(1 + \frac{p}{1-p} \right) - \frac{1}{2\sigma^2} \left(\ln \left(\frac{p}{1-p} \right) - \mu \right)^2 \right\}}{p(1-p) \sqrt{2\pi} \gamma_0 \exp \left\{ \eta_0 \sum_{i=1}^k y_i - nk \ln (1 + \exp(\eta_0)) - \frac{1}{2\sigma^2} (\eta_0 - \mu)^2 \right\}}$$

A Figura (1) apresenta uma comparação entre o resultado obtido pelo Método de Laplace e o resultado obtido pelo Método INLA que será estudado nas próximas seções. Em ambos os casos o valor estimado para a verdadeira proporção com base na amostra foi 0.3.

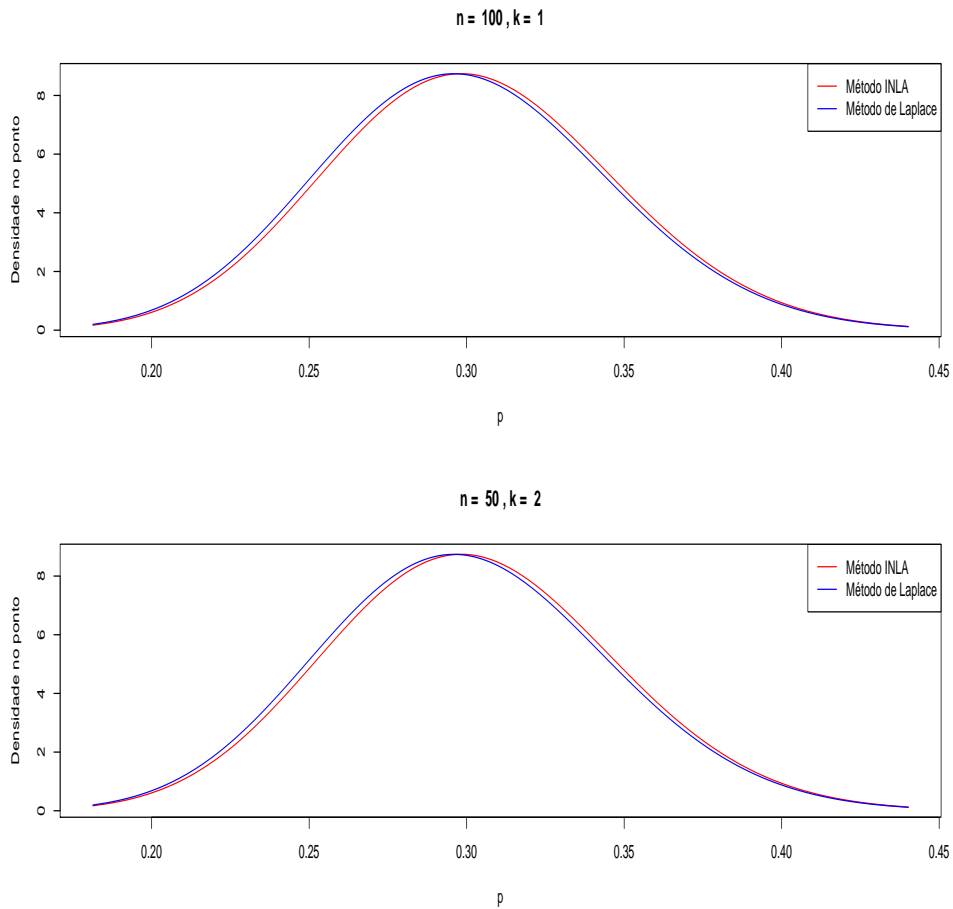


Figura 1: Comparação entre o Método de Laplace e o Método INLA para o modelo Binomial.

Observamos que as distribuições a posteriori obtidas pelos dois diferentes métodos são bastante similares. A seguir uma das aproximações determinísticas para distribuições a posteriori, a aproximação Gaussiana, será apresentada.

1.4 Aproximação Gaussiana

Dada uma função densidade de probabilidade da seguinte forma,

$$\pi(x) = \exp(f(x))$$

A aproximação gaussiana de $\pi(x)$ pode ser obtida da seguinte forma: Construa uma expansão de Taylor até segunda ordem de $f(x)$ em torno de x_0 ,

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)(x - x_0)^2}{2}$$

$$\propto b(x_0)x - \frac{1}{2}c(x_0)x$$

onde $b(x_0) = f'(x_0) - f''(x_0)x_0$ e $c(x_0) = -f''(x_0)$

Agora aplicando novamente a função exponencial temos que,

$$\tilde{\pi}_G(x) \approx \exp \left\{ -\frac{1}{2}c(x_0)x + b(x_0)x \right\} \quad (4)$$

onde $\tilde{\pi}_G(x)$ é uma aproximação gaussiana para $\pi(x)$.

Interessante se observar que esta expressão é o núcleo de uma distribuição Normal com média $\frac{b(x_0)}{c(x_0)}$ e variância $c(x_0)^{-1}$.

Estudos mostram que o ponto x_0 que produz as melhores aproximações se dá quando x_0 é igual a moda da distribuição $\exp(f(x))$. [8]

Para exemplificar suponha Y uma realização de uma distribuição Binomial, onde em $n = 100$ tentativas observamos 50 sucessos e suponha priori $\text{Beta}(a, b)$ para o parâmetro p . Desta forma calcula-se algebricamente a posteriori para o parâmetro p , que será a distribuição $\text{Beta}(y+a, n-y+b)$. Ou seja, a distribuição a posteriori exata para p é conhecida.

Então temos que,

$$L(y; p) \propto p^y(1-p)^{n-y}$$

e

$$\pi(p) \propto p^{a-1}(1-p)^{b-1}$$

Combinando as duas distribuições chegamos em uma distribuição proporcional a posteriori de p ,

$$\pi(p|y) \propto p^{y+a-1}(1-p)^{n-y+b-1}$$

Uma priori não informativa nesse caso é obtida fazendo-se $a = b = 1$. Nesse caso temos então,

$$\pi(p|y) \propto p^y(1-p)^{n-y}$$

Para aplicar a aproximação Gaussiana precisamos deixar esta expressão da seguinte forma,

$$\pi(p|y) \propto \exp \{y \log(p) + (n-y) \log(1-p)\}$$

Desta forma temos que $f(p) = y \log(p) + (n - y) \log(1 - p)$. Podemos utilizar a proporcionalidade na expressão acima, pois mesmo caso hajam constantes em $f(p)$, ao derivarmos a expressão essas constantes desapareceriam, ou seja, constantes não irão alterar o valor de $b(p_0)$ e $c(p_0)$.

Podemos agora aplicar a expansão de Taylor em torno de p_0 , para isso tomamos a primeira e a segunda derivadas de $f(p)$.

$$f'(p) = \frac{y}{p} + \frac{n - y}{p - 1}$$

e

$$f''(p) = -\frac{y}{p^2} - \frac{n - y}{(p - 1)^2}$$

precisamos agora encontrar as quantidades $b(p_0)$ e $c(p_0)$ como segue,

$$b(p_0) = f'(p_0) - f''(p_0)p_0 = \frac{y}{p_0} + \frac{n - y}{p_0 - 1} + \frac{y}{p_0^2} + \frac{n - y}{(p_0 - 1)^2}$$

e

$$c(p_0) = -f''(p_0) = +\frac{y}{p_0^2} + \frac{n - y}{(p_0 - 1)^2}$$

Então sabemos que $\pi(p|y) \approx N\left(\frac{b(p_0)}{c(p_0)}, \frac{1}{c(p_0)}\right)$

Podemos comprovar na Figura (2) que a melhor aproximação se dá quando escolhemos pontos p_0 próximos da moda da distribuição (que neste caso é a própria proporção de sucessos amostral, 0,5) para executar a aproximação.

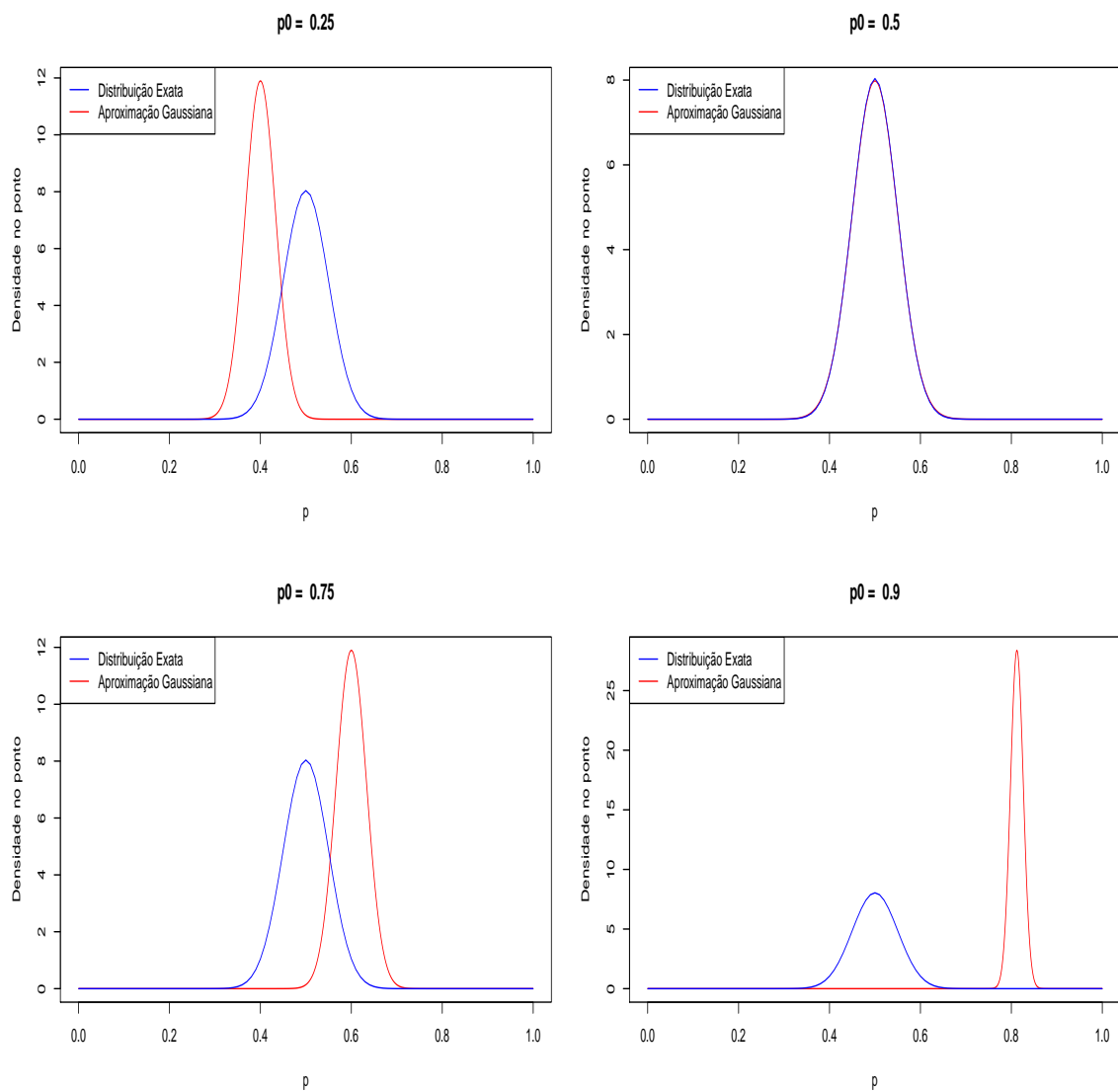


Figura 2: Aproximação gaussiana para a distribuição a posteriori do parâmetro p de uma Binomial.

2 Introdução ao método INLA

Como já mencionado, em Inferência Bayesiana, o maior interesse está na obtenção das distribuições a posteriori marginais para os parâmetros desconhecidos do modelo. Porém, em muitos casos a obtenção analítica de tais distribuições se torna demasiadamente complexa ou até mesmo impossível.

No ano de 2009, Rue H., Martino S. and Chopin N. apresentaram uma nova metodologia para aproximações de distribuições marginais a posteriori no trabalho *Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximation* [9]. A grande vantagem do método Aproximações de Laplace Aninhadas Integradas (do inglês, *Integrated Nested Laplace Approximations* – INLA) é o seu tempo computacional significativamente menor que o tempo computacional de métodos de simulação, tais como MCMC, além de não ter a necessidade de verificação de convergência, já que o INLA é um método determinístico.

O método INLA permite a aproximação das distribuições marginais a posteriori de modelos estruturados aditivos. Estes modelos formam uma ampla classe de modelos onde temos suposições mais flexíveis do que, por exemplo, a classe dos Modelos Lineares Generalizados (do inglês Generalized Linear Models - GLM). Em alguns tipos de modelagens os Modelos Lineares Generalizados não são adequados, pois podemos ter violações em algumas de suas suposições. Nesta direção a classe de modelos de regressão estruturados aditivamente tem sido útil, justamente devido a sua flexibilidade. Caso seja de interesse saber mais sobre estes modelos, pode-se consultar [1] ou então [4].

Para definir os modelos estruturados aditivos, considere uma variável resposta Y_i , $i = 1, \dots, n$. Digamos que a distribuição de Y_i pertença a família exponencial e que estamos interessados em um parâmetro μ_i que representa a média desta distribuição. Ao utilizarmos uma função de ligação $g(\cdot)$ temos um preditor $\eta_i = g(\mu_i)$, onde η pode ser escrito como,

$$\eta_i = \alpha + \sum_{j=1}^{nf} f^{(j)}(u_{ji}) + \sum_{k=1}^{n\beta} \beta_k z_{ki} + \epsilon_i \quad (5)$$

onde α é uma constante, $f(u_{ji})$ são funções desconhecidas das covariáveis u do modelo, β_k são efeitos lineares das covariáveis z e ϵ_i são termos não estruturados [1].

Uma grande variedade de modelos se enquadra nesta classe de modelos. Por exemplo, em modelos de regressão linear múltipla, a distribuição de Y é normal, as covariáveis z são as variáveis independentes, e temos apenas os parâmetros α e β_k como desconhecidos. Em modelos mistos generalizados, os parâmetros desconhecidos são α , β_k e também $f(u_{ji})$, onde u_{ji} serão os efeitos aleatórios e f será considerada uma função identidade.

Quando assumimos priori gaussiana para todos os parâmetros que compõe η , o vetor desses parâmetros é chamado de Campo Latente Gaussiano. E assim têm-se os Modelos Latentes Gaussianos.

Para a utilização do método INLA, os modelos Modelos Estruturados Aditivos são escritos como um modelo hierárquico bayesiano, com hierarquia definida da seguinte forma:

$$\pi(x, \theta|y) \propto \pi(\theta)\pi(x|\theta) \prod_{i=1}^n \pi(y_i|x_i, \theta) \quad (6)$$

onde Y é o vetor de valores da variável resposta obtidos na amostra; x é um vetor que contém todos os parâmetros desconhecidos em η , isto é, é o campo latente gaussiano; θ é um vetor de hiperparâmetros, isto é, parâmetros da priori gaussiana do campo latente; $\pi(x, \theta|y)$ é a distribuição a posteriori conjunta para os vetores x e θ , $\pi(\theta)$ é a priori atribuída ao vetor θ , $\pi(x|\theta)$ é a distribuição a priori de x dado o vetor de hiperparâmetros θ e $\prod_{i=1}^n \pi(y_i|x_i, \theta)$ a função de verossimilhança.

A ideia do método INLA é aproximar as seguintes distribuições de interesse:

$$\pi(x_i|y) \quad e \quad \pi(\theta_j|y)$$

Tais posterioris marginais por definição podem ser obtidas da seguinte forma:

$$\pi(x_i|y) = \int \pi(x_i|\theta, y)\pi(\theta|y)d\theta \quad (7)$$

e

$$\pi(\theta_j|y) = \int \pi(\theta|y) d\theta_{-j} \quad (8)$$

Como estas integrais em geral não são resolvíveis analiticamente, uma solução encontrada é aproxima-las. Para isso utiliza-se uma grade de valores primeiramente para θ e em um segundo momento para o vetor x , e então as integrais são aproximadas por soma finita. A grade pode ser obtida escolhendo valores ao redor dos pontos de maior densidade das distribuições em questão. Para mais detalhes sobre a obtenção da grade de valores ver, por exemplo, Seção 2.3 de [8] ou então [9]. Após escolher os pontos para formar as grades, aproximamos as equações (7) e (8) da seguinte forma,

$$\tilde{\pi}(x_i|y) = \sum_k \tilde{\pi}(x_i|\theta_k, y)\tilde{\pi}(\theta_k|y)\Delta_k \quad (9)$$

e

$$\tilde{\pi}(\theta_j|y) = \sum_k \tilde{\pi}(\theta_k|y) \Delta_{jk} \quad (10)$$

Como estaremos realizando uma soma finita não estaremos fazendo o cálculo exato e sim aproximações. Abaixo mostraremos como aproximar as distribuições contidas em (9) e (10).

2.1 Aproximação para $\pi(\theta|y)$

Sabe-se que

$$\pi(\theta|y) = \frac{\pi(x, \theta|y)}{\pi(x|\theta, y)} \quad (11)$$

A proposta do método INLA é fazer a seguinte aproximação:

$$\tilde{\pi}(\theta|y) \propto \frac{\pi(x, \theta, y)}{\pi_G(x|\theta, y)} \Bigg|_{x=x(\theta)} \quad (12)$$

onde, $x(\theta)$ é a moda obtida da distribuição $\pi(x|\theta, y)$ através de algum método de otimização. Note que para cada valor de θ teremos um $x(\theta)$.

O denominador de (12) é uma aproximação gaussiana para a condicional completa $\pi(x|\theta, y)$, como descrito na Seção (1.3).

$\pi(\theta|y)$ em geral não é gaussiana, então aplicar diretamente aproximação gaussiana em $\pi(\theta|y)$ não parece interessante. Porém, como assumimos distribuição gaussiana a priori para o vetor x então faz sentido utilizar aproximação gaussiana no denominador de $\pi(\theta|y)$. A expressão (12) é equivalente ao Método de Laplace descrito na Seção (1.3). [9]

2.2 Aproximação para $\pi(x_i|\theta, y)$

Rue et al. [9] propõe três formas de aproximação para $\pi(x_i|\theta, y)$.

- **Aproximação Gaussiana:** É a mais rápida de ser obtida e oferece resultados satisfatórios, porém pode apresentar erros de locação e/ou erros devido à falta de assimetria da aproximação. [8]
- **Aproximação de Laplace:** Oferece resultados muito precisos e é exata quando a condicional completa $\pi(x|\theta, y)$ for gaussiana (o que não ocorre quando a verossimilhança não é gaussiana), porém é muito custosa computacionalmente. [8]
- **Aproximação de Laplace Simplificada:** É a aproximação utilizada pelo método *R-INLA*. Basicamente faz-se uma melhoria na Aproximação de Laplace convencional para

corrigir alguns problemas computacionais, de forma que os resultados mantenham-se parecidos porém de uma forma muito mais rápida. [8]

2.3 *R-INLA*

Toda a metodologia utilizada no INLA está implementada no pacote *R-INLA*, que pode ser obtido pelo comando,

```
source("http://www.math.ntnu.no/inla/givemeINLA.R").
```

Como material introdutório podemos citar os exemplos disponíveis em <http://www.r-inla.org/> ou então o seguinte tutorial statistica.it/gianluca/Talks/INLA.pdf.

O comando básico para ajustar um modelo através deste método é:

```
resultado = inla(formula,data=data,family='`family`')
```

onde o argumento *family* indica a distribuição dos dados, *data* é o conjunto de dados a ser utilizado e *formula* é onde serão indicados os componentes do modelo. A maneira como se estabelece essa fórmula é similar a notação do modelo estruturado aditivo dado na Equação (5). Nas próximas subsessões alguns exemplos serão fornecidos. Os modelos para os dados observados (isto é, as funções de verossimilhança) possíveis de serem ajustados estão listados em www.r-inla.org/models/likelihoods, bem como a lista de possíveis modelos para os efeitos aleatórios em www.r-inla.org/models/latent-models e possíveis priors para os hiperparâmetros em www.r-inla.org/models/priors. Em <http://www.r-inla.org/examples> estão disponíveis vários exemplos, inclusive alguns retirados do manual de exemplos do WinBUGS, conhecido aplicativo computacional de MCMC.

Utilizando o comando *names(resultado)*, temos uma lista de informações adicionais, apresentadas no Quadro (1). Com essas opções podemos acessar muitas informações a respeito das distribuições a posteriori dos parâmetros e dos hiperparâmetros do modelo. Cada uma pode ser acessada colocando-se *resultado\$* antes do nome.

names.fixed	summary.fixed	marginals.fixed	summary.lincomb	marginals.lincomb
size.lincomb	summary.lincomb.derived	marginals.lincomb.derived	size.lincomb.derived	mlik
cpo	po	model.random	summary.random	marginals.random
size.random	summary.linear.predictor	marginals.linear.predictor	summary.fitted.values	marginals.fitted.values
size.linear.predictor	summary.hyperpar	marginals.hyperpar	internal.summary.hyperpar	internal.marginals.hyperpar
si	offset.linear.predictor	model.spde2.blc	summary.spde2.blc	marginals.spde2.blc
size.spde2.blc	logfile	misc	dic	mode
neffp	joint.hyper	nhyper	version	Q
graph	ok	cpu.used	.args	call
model.matrix				

Quadro 1: Opções de saídas utilizando o comando “inla”.

Outro aspecto interessante do INLA é a facilidade da geração de gráficos. Com o seguinte comando podemos facilmente solicitar os gráficos de interesse (Efeitos fixos, Efeitos aleatórios e Hiperparâmetros),

```
plot(resultado, plot.fixed.effects = TRUE, plot.random.effects =
      FALSE, plot.hyperparameters = FALSE)
```

Caso queiramos o gráfico dos efeitos aleatórios e dos hiperparâmetros (quando existem no modelo), basta substituir a palavra “FALSE” por “TRUE” em *plot.random.effects* e *plot.hyperparameters*.

Dentro do parâmetro *formula* podemos acrescentar diversas estruturas para os dados. Por exemplo, um modelo de regressão linear simples ($y_i = \beta_0 + \beta_1 x_i + \epsilon_i$) pode ser incorporado da seguinte forma:

```
formula= y ~ 1 + var1
```

onde *var1* é a variável explicativa x_i .

2.4 Exemplos

Esta subseção destina-se a ilustração de aspectos matemáticos envolvidos no método INLA. Escolhemos dois exemplos simples para mostrar parte da matemática inserida no método e fazer sua implementação no software R, em seguida foi realizada a comparação entre o resultado trazido pelo *R-INLA* e pela implementação feita.

Por facilidade de cálculos e por ter resultados satisfatórios para modelos simples, utilizaremos a aproximação gaussiana sempre que se fizer necessário o uso de aproximações.

2.4.1 Modelo Binomial

No modelo Binomial encontramos a seguinte situação. Dado Y_1, Y_2, \dots, Y_k uma amostra aleatória de tamanho k de uma população com distribuição Binomial(n, p), onde n é conhecido, nosso interesse está em descobrir a distribuição a posteriori de p .

Para utilizar a metodologia de Inferência Bayesiana, podemos primeiramente atribuir uma distribuição de probabilidade a priori para o parâmetro desconhecido p , contendo a informação prévia que temos a respeito deste parâmetro.

Utilizando as ideias do método INLA, vemos que neste modelo temos apenas um parâmetro desconhecido, portanto temos apenas o vetor $x = \{p\}$ (Ver Equação (6)), precisamos aplicar uma priori Normal para o vetor x , o que não é possível para este parâmetro em questão, pois, seu espaço paramétrico é limitado ($p \in [0, 1]$). Para contornar esse problema podemos fazer uma transformação no parâmetro p como segue,

$$\eta = \log\left(\frac{p}{1-p}\right) = x \quad (13)$$

Agora o espaço paramétrico de η é dado pelos números reais ($\eta \in \mathfrak{R}$), e podemos atribuir portanto uma priori Normal para este parâmetro. Note que agora o nosso parâmetro p pode ser reescrito como sendo $\frac{\exp(\eta)}{1 + \exp(\eta)}$. Tal transformação também vai de encontro com a necessidade de adequar o modelo à classe dos modelos aditivos generalizados. Assim, estamos usando a função de ligação logito e considerando que o único parâmetro desconhecido em (5) é o intercepto $\alpha = \eta$.

No Quadro (2) temos códigos do R para gerar dados conforme o modelo Binomial e analisá-los utilizando os comandos já implementados do INLA. Nas seis primeiras linhas, fixamos uma semente aleatória para que os mesmos resultados possam ser repetidos e geraremos uma amostra de tamanho 50 de uma população Binomial com $n = 1$ e $p = 0.3$. O parâmetro n pode ser interpretado como o número de ensaios e o parâmetro p pode ser interpretado como a probabilidade de obtermos sucesso em cada um destes n ensaios. Ao fixar $n = 1$, temos a situação bastante comum onde Y são variáveis dicotômicas, ou seja, com distribuição Bernoulli.


```

1  p=0.3
2  n=1
3  k=50
4  set.seed(4 + 8 + 15 + 16 + 23 + 42)
5  y = rbinom(k,n,p)
6  data=data.frame(y)

7  sigma2=10000
8  precisao=1/sigma2

9  library(INLA)
10 formula=y~1
11 resultado=inla(formula,data=data,family="binomial",Ntrials=n,
    control.fixed = list(prec.intercept = precisao))

```

Quadro 2: Programações iniciais para o modelo Binomial.

Nas linhas 7 e 8 do Quadro (2) definimos $\tau = \frac{1}{\sigma^2}$. Desta forma $\eta \sim Normal\left(0, \frac{1}{\tau}\right)$. Esta parametrização foi realizada desta forma pois o padrão do *R-INLA* é utilizar priori com média 0 e precisão τ para o os efeitos fixos do modelo.

Nas últimas três linhas do Quadro (2) montamos o modelo. Através da *formula* estamos informando ao método que a nossa variável Y é explicada por uma constante (equivalente ao α da Equação (5)), em *resultado* estamos informando a estrutura do modelo (*formula*), em que variável estão os dados (*data*), a qual família pertencem estes dados pertencem (*family*), o parâmetro n do modelo Binomial (*Ntrials*) e através do comando *control.fixed* estamos informando a precisão a priori de η . A função de ligação logito já é padrão ao se especificar a família binomial.

Antes de apresentarmos os resultados iremos mostrar os passos matemáticos para repetir o que estas linhas de comandos acabaram de fazer. Os comandos estão apresentados no Quadro (3).

```

1  pig.eta_y = function(eta,etam,n,y,precisao)
2  {
3  k = length(y)
4  feta = etam*sum(y) - n*k*log(1+exp(etam)) -(etam**2)*precisao/2
5  fleta = sum(y) -n*k*exp(etam)/(1+exp(etam)) -etam*precisao
6  fleta = -n*k*exp(etam)/((1+exp(etam))**2) -precisao
7  bx0 = fleta-fleta*etam
8  cx0 = - fleta
9  dnorm(eta, mean=bx0/cx0, sd=sqrt(1/cx0))
10 }

11 pi.eta_y=function(x)
12 {
13 -exp(x*sum(y) - n*k*log(1+exp(x)) -(x**2)*precisao/2)
14 }

15 OP=optimize(pi.eta_y,c(-10,10))
16 em=OP$minimum

17 valor.eta=pig.eta_y(eta,em,n,y,precisao)

18 bases = rep(0,length(eta))

19 for(i in 2:(length(eta)-1))
20 {
21 bases[i] = mean(c(eta[i],eta[i+1])) - mean(c(eta[i],eta[i-1]))
22 }

23 bases[1] = mean(bases[-c(1,length(eta))])
24 bases[length(eta)] = bases[1]
25 aux1=bases*valor.eta
26 const=sum(aux1)

```

Quadro 3: Programações utilizadas no modelo Binomial.

Estamos interessados em obter,

$$\pi(x|y)$$

onde x é o nosso vetor de parâmetros desconhecidos, neste caso $x = \eta$. Temos também que,

$$Y_i \sim \text{Binomial} \left(n, \frac{\exp(\eta)}{1 + \exp(\eta)} \right), i = 1, 2, \dots, k$$

Então a função de verossimilhança dos dados é dada por:

$$L(y; \eta) = \prod_{i=1}^k C_{y_i}^n \frac{\exp(\eta)^{y_i}}{(1 + \exp(\eta))^{y_i}} \left(1 - \frac{\exp(\eta)}{1 + \exp(\eta)}\right)^{(n-y_i)} \quad (14)$$

Como priori para η estamos usando a já mencionada distribuição Normal $\left(0, \frac{1}{\tau}\right)$, com densidade:

$$\pi(\eta) = \frac{1}{\sqrt{2\pi}} \tau \exp\left\{\frac{-1}{2} \tau \eta^2\right\} \quad (15)$$

Combinando as equações (14) e (15) chegaremos na distribuição a posteriori para η ,

$$\pi(\eta|y) \propto L(y; \eta)\pi(\eta) \quad (16)$$

Fazendo algumas simplificações chegamos em,

$$\pi(\eta|y) \propto \exp\left\{\sum_{i=1}^k y_i \eta - nk \log(1 + \exp(\eta)) - \frac{\eta^2}{2} \tau\right\} \quad (17)$$

Neste ponto o INLA faz uma das aproximações descritas na Seção (2.3). Como já mencionado vamos adotar a aproximação Gaussiana (descrita na Seção (1.3)) para a Equação 17.

Para utilizar essa aproximação precisamos que a equação esteja na forma $\pi(\eta) \propto \exp(f(\eta))$, o que já está atendido na Equação (17), onde $f(\eta)$ é dada por,

$$f(\eta) = \sum_{i=1}^k y_i \eta - nk \log(1 + \exp(\eta)) - \frac{\eta^2}{2} \tau \quad (18)$$

Precisamos agora aplicar a aproximação de Taylor até segunda ordem em (18), para isso precisamos da primeira e segunda derivadas de (18).

$$f'(\eta) = \sum_{i=1}^k y_i - nk \frac{\exp(\eta)}{1 + \exp(\eta)} - \eta \tau \quad (19)$$

$$f''(\eta) = -nk \frac{\exp(\eta)}{(1 + \exp(\eta))^2} - \tau \quad (20)$$

Desta forma, podemos dizer que,

$$f(\eta) \approx f(\eta_0) + f'(\eta_0)(\eta - \eta_0) + f''(\eta) \frac{(\eta - \eta_0)^2}{2} \quad (21)$$

onde η_0 é a moda da distribuição $\pi(\eta|y)$.

O nosso interesse está na parte da expressão (21) que envolve η , desta forma podemos simplificar a expressão (21) e obter,

$$f(\eta) \approx b(\eta_0)\eta - \frac{1}{2}c(\eta_0)\eta^2 \quad (22)$$

onde, $b(\eta_0) = f'(\eta_0) - f''(\eta_0)\eta_0$ e $c(\eta_0) = -f''(\eta_0)$.

Aplicando a função exponencial chegamos então na aproximação gaussiana para $\pi(\eta|y)$,

$$\pi_G(\eta|y) \approx \exp \left\{ -\frac{1}{2}c(\eta_0)\eta^2 + b(\eta_0)\eta \right\} \quad (23)$$

Vemos que a Equação (23) é o núcleo de uma distribuição Normal com média $\frac{b(\eta_0)}{c(\eta_0)}$ e precisão $c(\eta_0)$. Tal aproximação está programada nas linhas 1 a 10 do Quadro (3).

Neste ponto já temos a aproximação para a distribuição a posteriori de η , basta encontrarmos o ponto η_0 onde será feita a aproximação de Taylor. Este ponto é encontrado otimizando a função (17) com algum método computacional como por exemplo utilizando a função *optimize* do R. Esta otimização se encontra nas linhas 11 a 16 do Quadro (3)

Tendo a aproximação da distribuição a posteriori podemos então avaliá-la em uma grade de pontos para η e então observar seu comportamento. Por exemplo, se a grade de pontos escolhida for 1000 pontos igualmente espaçados entre -2 e 0, teremos o gráfico apresentado na Figura (3) para $\pi_G(\eta|y)$,

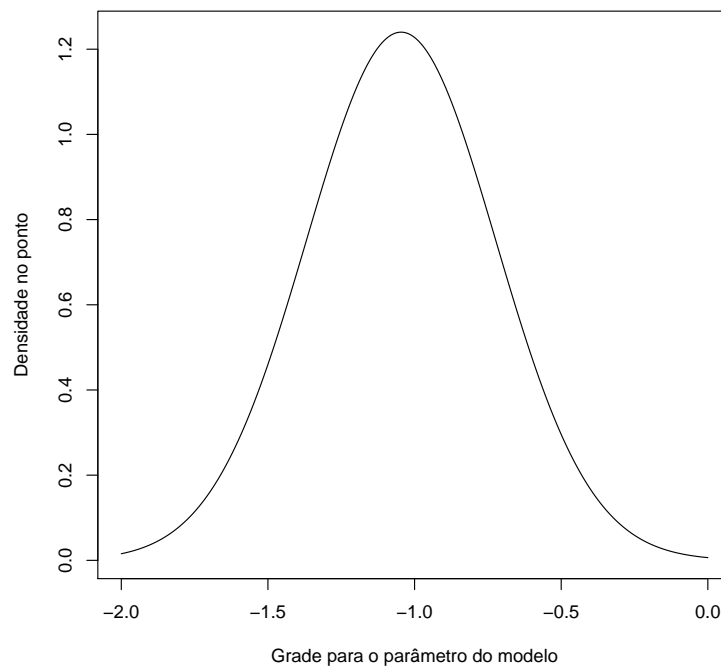


Figura 3: Distribuição a posteriori para o logito da probabilidade de sucesso no modelo Binomial utilizando uma grade contendo 1000 pontos.

Para podermos comparar resultados utilizamos a mesma grade de pontos utilizada pelo *R-INLA* que é diferente da grade utilizada no gráfico da Figura (3).

Neste ponto devemos atentar para o fato de que nós derivamos uma função que é proporcional à $\pi(\eta|y)$ e não nos preocupamos com o fato de que a distribuição de probabilidade deve integrar 1 em todo seu espaço paramétrico.

Como podemos ver na Equação (23) este resultado não define uma distribuição de probabilidade, devemos então encontrar a constante normalizadora para o modelo.

Como estamos utilizando uma grade de pontos, uma maneira simples de encontrar a constante normalizadora é descobrindo o valor das bases (espaços entre os valores gerados pela grade) e multiplicar pelo valor da função neste ponto para todos os pontos da grade. Feito isso somamos todos os valores obtidos e encontramos a constante normalizadora (linhas 18 a 26 do Quadro (3)). Em seguida, basta dividir o valor da função em cada um dos pontos por essa constante, garantido que a área abaixo da curva seja 1. O valor da função em cada um dos pontos está sendo obtido na linha 17 do Quadro (3).

Para uma comparação gráfica da aproximação utilizada pelo *R-INLA*, a distribuição exata e a aproximação Gaussiana para esse modelo, faremos nove cenários alternando o valor do parâmetro desconhecido, o número de ensaios e também o tamanho de amostra. A grade utilizada foi a mesma utilizada pelo *R-INLA* em sua programação. Os valores foram escolhidos de forma que tivéssemos uma grande variedade de combinações de k , n e verdadeira proporção (p).

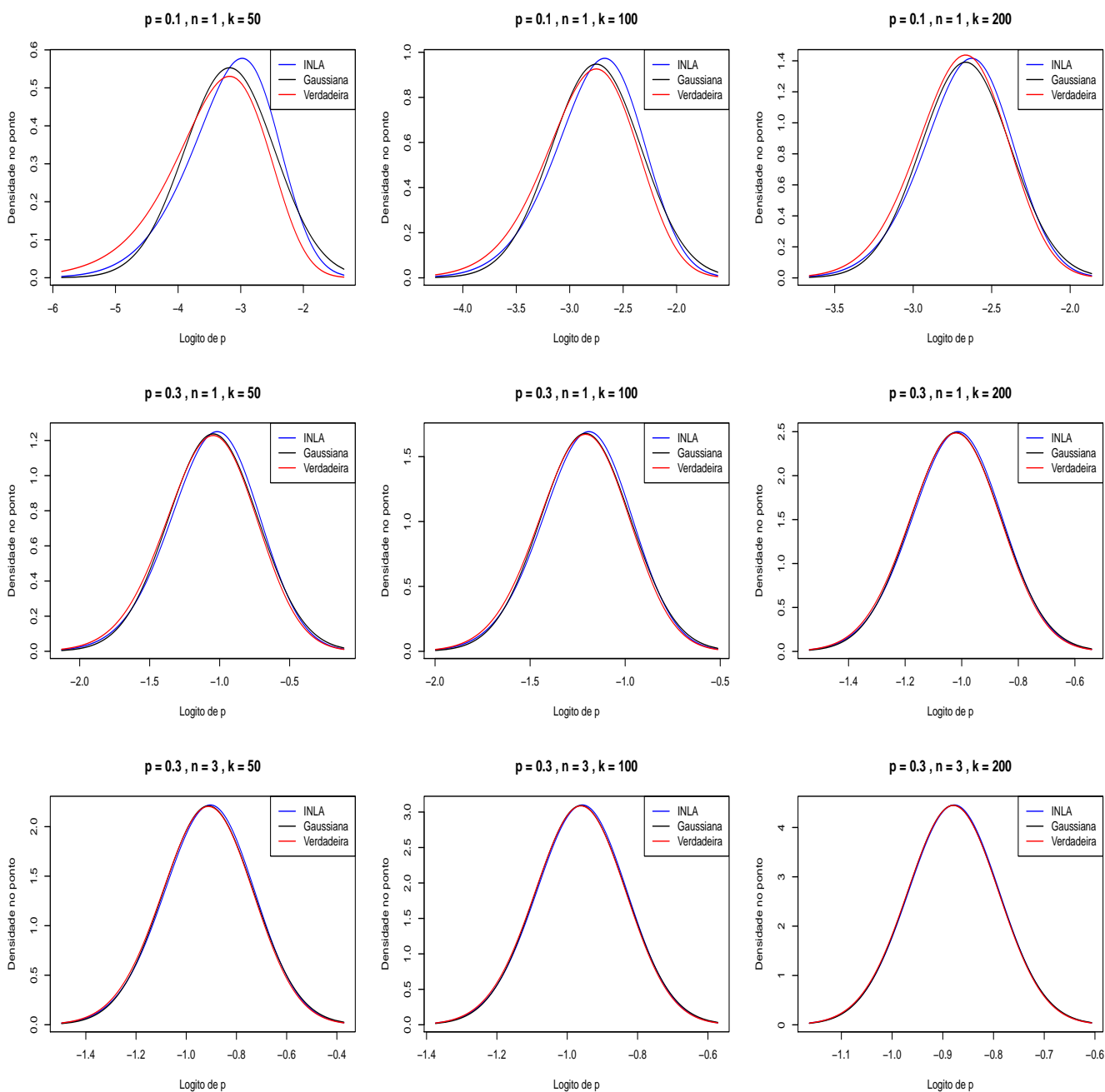


Figura 4: Distribuição a posteriori do parâmetro η para algumas combinações de k , n e verdadeira proporção (p).

Podemos ver na Figura (4) que para o caso onde o número de ensaios é 50 e o verdadeiro valor da proporção igual a 0,1 ocorrem as maiores dissimilaridades entre as curvas. Vemos também que ao aumentar o número de ensaios a curva traçada pelo *R-INLA* se aproxima da distribuição verdadeira, o mesmo vale para a aproximação Gaussiana.

Para fins de comparação com o resultado verdadeiro para p , necessitamos primeiramente derivar a distribuição a posteriori verdadeira de p analiticamente. Sabemos a distribuição a posteriori de η e também sabemos que p é função de η , então, pelo método Jacobiano podemos encontrar a distribuição de p .

$$\pi(\eta|y) = C \exp \left\{ \sum_{i=1}^k y_i \eta - nk \log(1 + \exp(\eta)) - \frac{\eta^2}{2} \tau \right\}$$

onde C é uma constante cujo valor é desconhecido. Utilizando uma grade de valores podemos aproximar essa constante, por exemplo, utilizando a função *integrate* do R.

Utilizando o método Jacobiano podemos derivar a distribuição de p . Após alguma algebra,

$$\pi(p|y) = \frac{C}{p(1-p)} \left(\frac{p}{1-p} \right)^{\sum_{i=1}^k y_i} \frac{1}{1-p} \left(\frac{p}{1-p} \right)^{\tau}$$

Agora sabemos qual a distribuição verdadeira para p utilizando priori Normal para η . A Figura (5) apresenta a comparação entre as posteriores utilizando priori Beta e priori Normal não informativas. Podemos ver que ao utilizarmos prioris não informativas teremos posteriores minimamente diferentes.

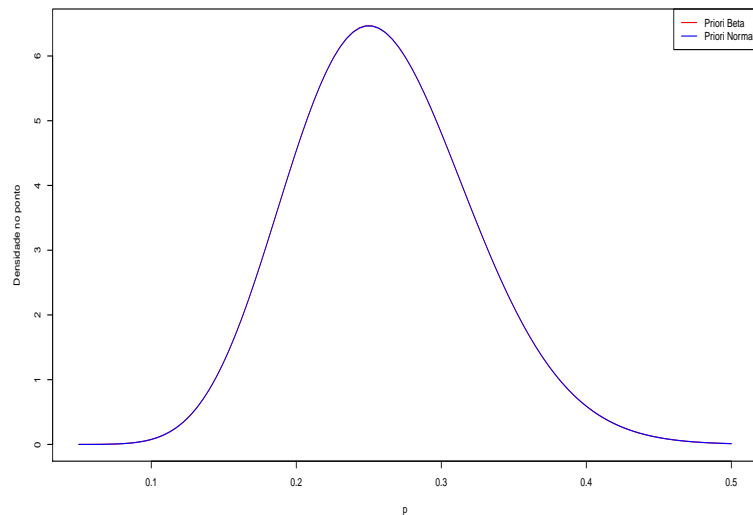


Figura 5: Comparação entre posteriores para o parâmetro p , utilizando priori Normal e Beta não informativas

Para uma comparação gráfica da aproximação utilizada pelo *R-INLA*, a distribuição exata e a aproximação Gaussiana para esse modelo, faremos nove cenários alternando o valor do parâmetro desconhecido, o número de ensaios e também o tamanho de amostra. A grade utilizada foi a mesma utilizada pelo *R-INLA* em sua programação. Os valores foram escolhidos de forma que tivéssemos uma grande variedade de combinações de k, n e verdadeira proporção (p).

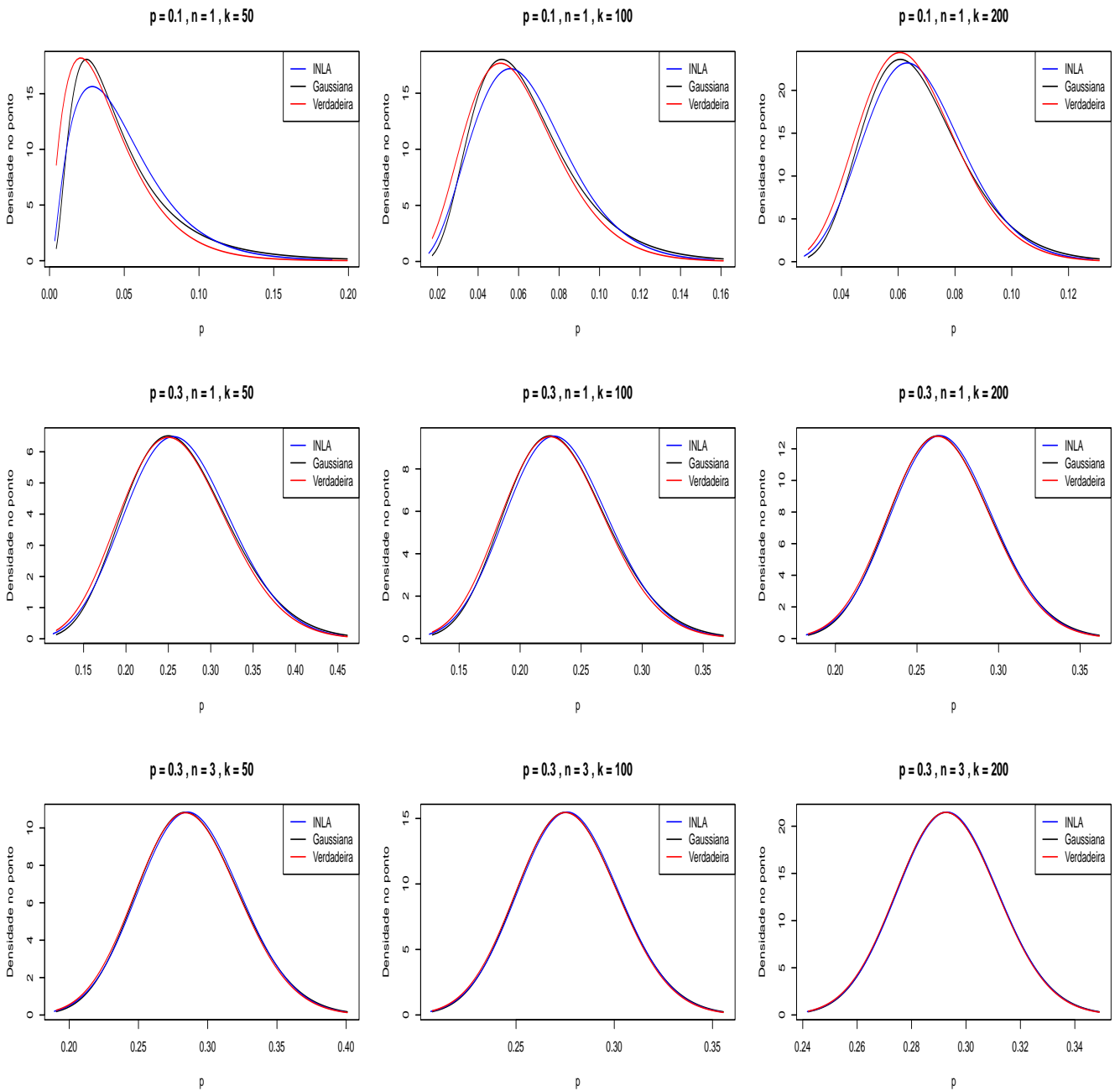


Figura 6: Distribuição a posteriori do parâmetro p para algumas combinações de k , n e verdadeira proporção (p).

Podemos ver na Figura (6) que para o caso onde o número de ensaios é 50 e o verdadeiro valor da proporção igual a 0,1 ocorrem as maiores dissimilaridades entre as curvas. Vemos também que ao aumentar o número de ensaios a curva traçada pelo *R-INLA* se aproxima da verdadeira, o mesmo vale para a aproximação Gaussiana.

É importante ressaltar que pela forma calculada chegaremos na posteriori para η e em seguida precisamos transformar essa distribuição para podermos comparar com a verdadeira distribuição de p . Essa transformação foi feita através da função *inla.tmargin* da seguinte forma,


```

a1=data.frame(x=eta,y=valor.eta)
marg.p.est=inla.tmarginal(function(eta) exp(eta)/(1+exp(eta)), a1)

```

2.4.2 Modelo Normal

No modelo Normal encontramos a seguinte situação. Dado Y_1, Y_2, \dots, Y_n uma amostra aleatória de tamanho n de uma população com distribuição Normal(μ, σ^2), nosso interesse está em encontrar as distribuições a posteriori marginais de μ e de σ^2 .

Neste modelo temos dois parâmetros desconhecidos, onde o vetor $x = \{\mu\}$ e o vetor $\theta = \{\sigma^2\}$. É interessante observar que tanto no caso do modelo Binomial quanto no modelo Normal a distribuição de x não depende de θ portanto na Equação (6) faz-se $\pi(x|\theta) = \pi(x)$.

No método INLA, o padrão é atribuir priori Log-gama para $\log\left(\frac{1}{\sigma^2}\right)$. Esta é a distribuição do logaritmo natural de uma variável aleatória com distribuição Gama (comumente utilizada como priori para a precisão τ). Sua densidade pode ser derivada através do método Jacobiano. Desta forma alteramos θ para:

$$\theta = \log\left(\frac{1}{\sigma^2}\right)$$

Nos Quadros (4) e (5) temos as programações utilizadas no R. No Quadro (4) estamos gerando uma amostra do modelo Normal e analisando-a utilizando os comandos já implementados do *R-INLA*. Nas primeiras seis linhas do Quadro (4), fixamos uma semente aleatória para que os mesmos resultados possam ser repetidos e geramos uma amostra de tamanho 50 de uma população Normal com $\mu = 10$ e $\sigma^2 = 4$. Esses valores posteriormente serão mudados para observarmos se há mudanças nos resultados.

```

1  mi=10
2  sigma=2
3  n=50
4  set.seed(4 + 8 + 15 + 16 + 23 + 42)
5  y = rnorm(n,mi,sigma)
6  data=data.frame(y)

7  a= 1
8  b= 5e-05
9  prec.mi=0.001

10 library(INLA)
11 formula=y~1
12 result=inla(formula,data=data,family="gaussian", control.fixed = list(prec.intercept=
    prec.mi),control.family=list(hyper = list(theta = list(prior="loggamma",param=c(a,b))))))

```

Quadro 4: Programações iniciais para o modelo Normal.

Nas linhas 7 a 9 do Quadro (4) definimos os parâmetros da Log-gama(a, b) (priori para θ) e definimos a precisão a ser utilizada na distribuição a priori $\pi(x|\theta) = \pi(x)$. Na linha 10 até a linha 12 analisamos o modelo através do INLA.

Em *control.family* estamos atribuindo uma priori Log-gama(a, b) para θ . Em *control.fixed* informamos a precisão a priori para $\pi(x|\theta)$. Em *data* informamos que os dados estão guardados na variável *data*. Através da *formula* informamos ao modelo que temos apenas o efeito fixo e em *family* informamos ao modelo que os dados seguem distribuição Normal.

Antes de apresentarmos os resultados iremos mostrar os passos matemáticos para repetir o que estas linhas de comandos acabaram de fazer. No Quadro (5) tem-se algumas das programações utilizadas para este modelo. Outras funções utilizadas durante a programação podem ser encontradas em anexo ao final da monografia. Novamente usamos a grade de valores obtida do próprio *R-INLA*.

```

1  aux5=rep(0,length(teta))
2  valor.teta=rep(0,length(teta))

3  for(i in 1:length(teta))
4  {
5  aux2=teta[i]
6  aux3=lp.mi
7  pi.x_gamay=function(x)
8  {
9  aux4=sum((y-x)**2)
10 - exp(n*aux2/2 - (exp(aux2)/2)*aux4 + aux3/2 - exp(aux3)*(x**2)/2)
11 }
12 NR=nlm(pi.x_gamay,10)
13 aux5[i]=NR$estimate
14 valor.teta[i] = pi.gama.post(aux2,a,b,aux5[i],y,aux3)
15 }
16 xm=aux5

17 bases = rep(0,length(teta))
18 for(i in 2:(length(teta)-1))
19 {
20 bases[i] = mean(c(teta[i],teta[i+1])) - mean(c(teta[i],teta[i-1]))
21 }
22 bases[1] = mean(bases[-c(1,length(teta))])
23 bases[length(teta)] = mean(bases[-c(1,length(teta))])
24 aux6=bases*valor.teta
25 aux6=sum(aux6)

26 media = marg.mi[,1]
27 valor.media =rep(0,length(media))

28 for(i in 1:length(media))
29 {
30 acm = 0
31 for(j in 1:length(teta))
32 {
33 acm = acm + pig.x_gamay(lp.mi,teta[j],xm[j],media[i],y)*pi.gama.post(teta[j],a,b,xm[j],y,lp.mi)
34 }
35 valor.media[i] = acm
36 }

37 bases = rep(0,length(media))
38 for(i in 2:(length(media)-1))
39 {
40 bases[i] = mean(c(media[i],media[i+1])) - mean(c(media[i],media[i-1]))
41 }
42 bases[1] = mean(bases[-c(1,length(media))])
43 bases[length(media)] = mean(bases[-c(1,length(media))])
44 aux7=bases*valor.media
45 aux7=sum(aux7)

```

Quadro 5: Programações utilizadas no modelo Normal.

Estamos interessados em obter informações para x e θ . Para isso precisamos definir algumas equações. A primeira delas é a verossimilhança dos dados,

$$L(y; x, \theta) = \prod_{i=1}^n \frac{\sqrt{\exp(\theta)}}{\sqrt{2\pi}} \exp \left\{ \frac{\exp(\theta)}{2} (y_i - x)^2 \right\} \quad (24)$$

A distribuição $\pi(x|\theta)$ será a distribuição Normal com média μ e variância $\sigma_x^2 = \frac{1}{\exp(\theta_x)}$,

$$\pi(x|\theta) = \frac{\sqrt{\exp(\theta_x)}}{\sqrt{2\pi}} \exp \left\{ \frac{-1}{2} \exp(\theta_x) \eta^2 \right\} \quad (25)$$

onde θ_x é um valor estipulado a priori para o logaritmo da precisão de x .

A priori para θ será a distribuição Log-gama(a, b),

$$\pi(\theta) = \frac{\exp(\theta)^a b^a}{\Gamma(a)} \exp \{-b \exp(\theta)\} \quad (26)$$

Sabemos da Seção (2.1) que para obtermos $\pi(\theta|y)$ precisamos primeiramente de $\pi(x|\theta, y)$, que pode ser obtida combinando as equações (24) e (25) como segue,

$$\pi(x|\theta, y) \propto L(y; x, \theta) \pi(x|\theta) \quad (27)$$

Fazendo algumas simplificações na Equação (27), chegamos em,

$$\pi(x|\theta, y) \propto \exp \left\{ -\frac{\exp(\theta)}{2} \sum_{i=1}^n (y_i - x)^2 - \frac{\exp(\theta_x)}{2} x^2 + \frac{n}{2} \theta + \frac{1}{2} \theta_x \right\} \quad (28)$$

Agora para utilizarmos a Equação (12) da Seção (2.1) é necessário fazer uma aproximação gaussiana para a Equação (28).

Para isso faremos uma aproximação de Taylor de segunda ordem em $f(x)$, onde,

$$f(x) = -\frac{\exp(\theta)}{2} \sum_{i=1}^n (y_i - x)^2 - \frac{\exp(\theta_x)}{2} x^2 + \frac{n}{2} \theta + \frac{1}{2} \theta_x \quad (29)$$

Então tomamos a primeira e a segunda derivadas da Equação (29) como segue,

$$f'(x) = \exp(\theta) \sum_{i=1}^n (y_i - x) - \exp(\theta_x) x \quad (30)$$

$$f''(x) = \exp(\theta) n - \exp(\theta_x) \quad (31)$$

Desta forma, podemos dizer que,

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + f''(x_0)\frac{(x - x_0)^2}{2} \quad (32)$$

onde x_0 é a moda da distribuição $\pi(x|\theta, y)$. Ou seja, para cada valor de θ teremos um x_0 diferente.

O nosso interesse está na parte da expressão (32) que envolve x , desta forma podemos simplificar a expressão (32) e obter,

$$f(x) \approx b(x_0)x - \frac{1}{2}c(x_0)x^2 \quad (33)$$

onde, $b(x_0) = f'(x_0) - f''(x_0)x_0$ e $c(x_0) = -f''(x_0)$.

Aplicando a função exponencial chegamos então na aproximação gaussiana para $\pi(x|\theta, y)$,

$$\pi_G(x|\theta, y) \approx \exp \left\{ -\frac{1}{2}c(x_0)x^2 + b(x_0)x \right\} \quad (34)$$

Vemos que a Equação (34) é o núcleo de uma distribuição Normal com média $\frac{b(x_0)}{c(x_0)}$ e precisão $c(x_0)$.

Neste ponto já temos a aproximação gaussiana para a distribuição $\pi(x|\theta, y)$, basta encontrarmos o ponto x_0 onde será feita a aproximação de Taylor. Este ponto é encontrado otimizando a função (27) com algum método computacional como por exemplo utilizando a função *optimize* ou então a função *nlm* do R, porém essa otimização deve ser feita para uma grade de θ onde para cada valor de θ tenhamos um ponto x_0 .

Sabemos da Equação (12) que a aproximação para $\pi(\theta|x)$ é dada por,

$$\tilde{\pi}(\theta|y) \propto \frac{\pi(x, \theta, y)}{\pi_G(x|\theta, y)} \Bigg|_{x=x(\theta)} \quad (35)$$

onde, $x(\theta)$ é obtida da distribuição $\pi(x|\theta, y)$ através de algum método de otimização. Esta otimização pode ser vista nas linhas 1 a 16 do Quadro (5)).

O numerador desta expressão pode ser reescrito da seguinte forma:

$$\pi(x, \theta, y) = \pi(\theta)\pi(x|\theta)L(y; x, \theta) \quad (36)$$

Agora temos todas as distribuições necessárias para se obter a posteriori para θ . Note que para cada valor da grade de θ aplicaremos a Equação (35) no ponto x_0 referente ao θ em questão. (Linha 14 do Quadro (5)).

Caso tivéssemos escolhido uma grade para θ com 100 valores igualmente espaçados entre -2 e 0 teríamos o gráfico apresentado na Figura (7) (na esquerda temos o gráfico sem a constante

normalizadora e na direita o gráfico com a constante normalizadora).

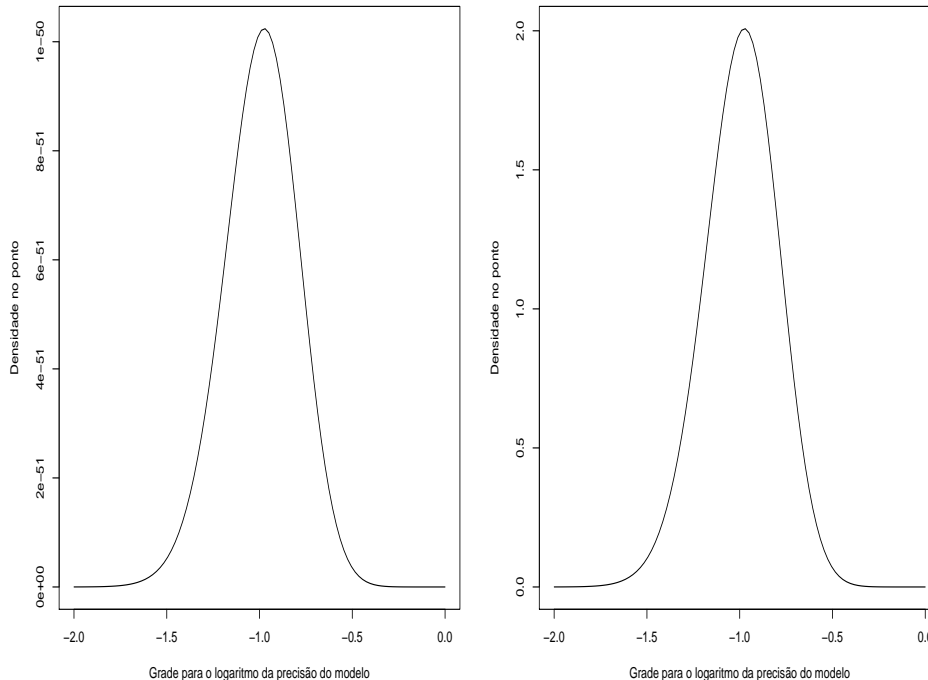


Figura 7: Distribuição a posteriori para o logaritmo da precisão do modelo Normal utilizando uma grade contendo 100 pontos.

Nas linhas 17 a 25 do Quadro (5), vemos como encontrar a constante normalizadora para esta distribuição.

Neste exemplo e utilizando a grade de 100 pontos, a constante normalizadora fará uma enorme diferença já que o valor encontrado para ela foi de $5.100443e-51$ que é bastante diferente de 1. Podemos ver que os valores no eixo y mudam muito de um gráfico para o outro, porém o formato da curva se mantém.

O próximo passo agora é obter a distribuição $\pi(x|y)$ que pode ser obtida de 3 formas como dito na Seção (2.2). Usaremos a aproximação gaussiana por ser de mais fácil implementação e por produzir resultados satisfatórios para modelos simples. Desta forma teremos,

$$\tilde{\pi}(x|y) = \sum_k \pi_G(x|\theta_{1k}, y) \tilde{\pi}(\theta_{1k}|y) \Delta_k \quad (37)$$

Neste caso precisamos realizar esta soma apenas em θ , para uma grade de valores. Entretanto essa soma deve ser realizada também para uma grade de valores para x, ou seja, para cada elemento da grade de x será feita a soma em uma grade para θ . (Linhas 28 a 36 do Quadro (5)).

Utilizando a mesma grade de θ já utilizada com 100 valores igualmente espaçados entre -2 e 0 e estipulando uma grade para x contendo 100 valores igualmente espaçados entre 9 e 11,

podemos obter a posteriori para x . A Figura (8) apresenta gráfico sem a constante normalizadora (esquerda) e na direita o gráfico com a constante normalizadora.

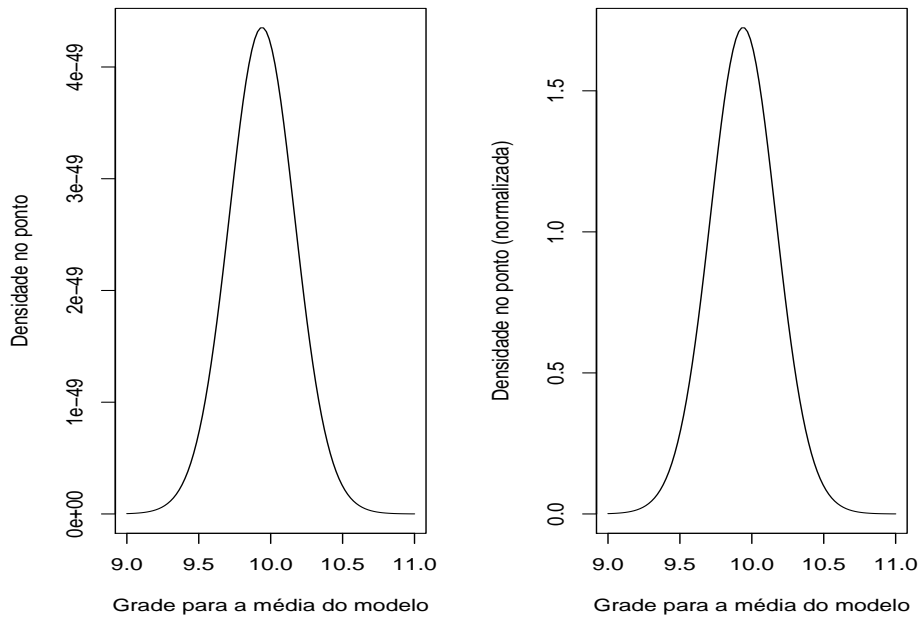


Figura 8: Distribuição a posteriori normalizada para a média do modelo Normal utilizando uma grade contendo 100 pontos.

Para uma comparação gráfica da aproximação utilizada pelo $R-INLA$ e a Aproximação Gaussiana, faremos quatro cenários alternando o valor da média e da variância do modelo Normal. Para geração do gráfico foi utilizada a mesma grade de valores utilizada pelo INLA.

Podemos ver que mesmo para o caso onde o tamanho de amostra é 5, a aproximação ficou bastante próxima do resultado calculado pelo $R-INLA$.

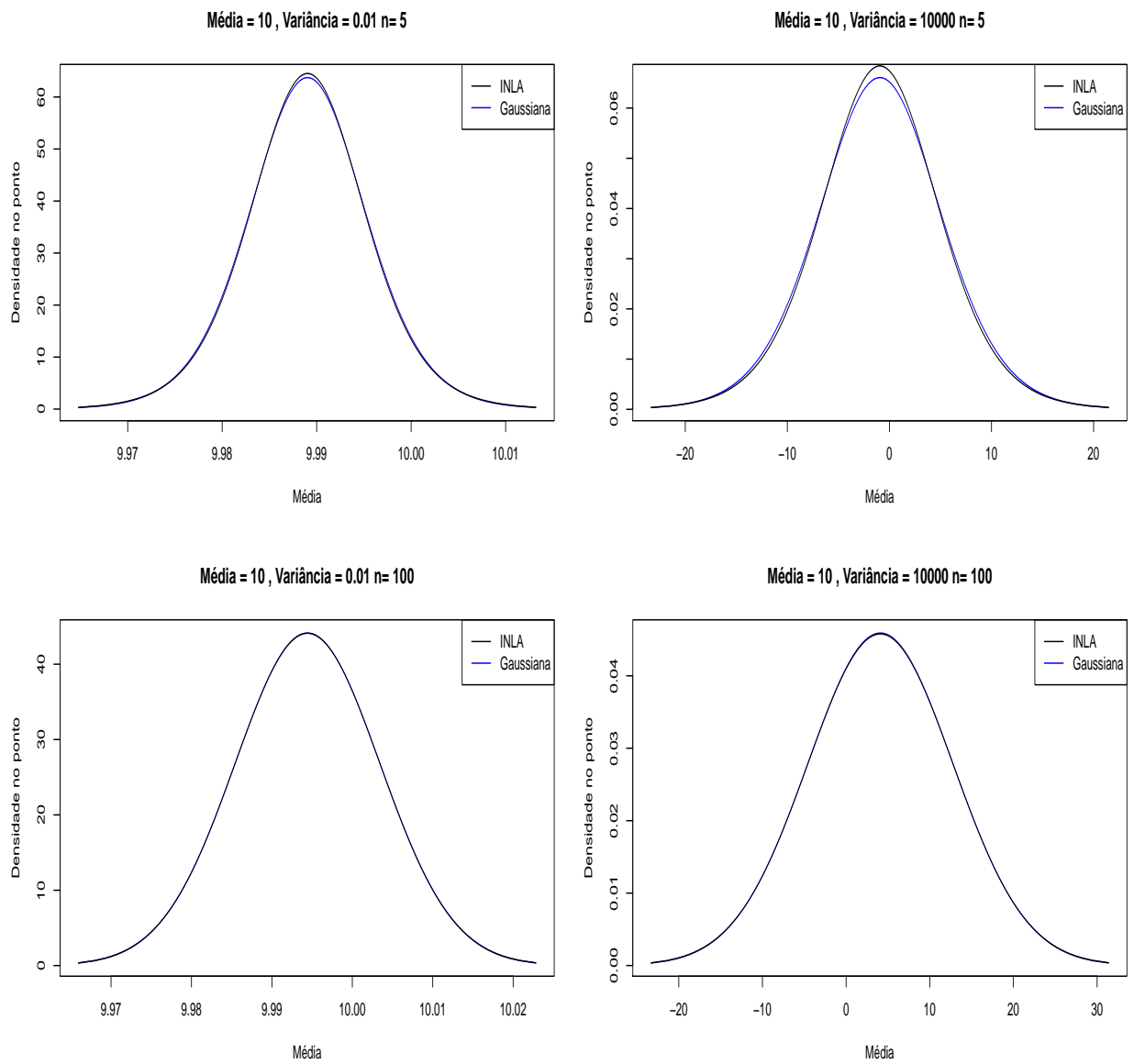


Figura 9: Distribuição a posteriori do parâmetro Média para algumas combinações de verdadeira Média, Variância e tamanho de amostra.

3 Aplicações

Através de três exemplos abordaremos os principais comandos do pacote *R-INLA*, com uma breve explicação dos mesmos.

O software WinBUGS (<http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/contents.shtml>) também será utilizado com a finalidade de comparar resultados e o tempo computacional. O computador utilizado opera com um processador AMD Phenom(tm) 8400 Triple-Core Processor 2.10 GHz e 2GB de memória RAM.

3.1 Regressão Logística com efeitos aleatórios

Para exemplificar o modelo de regressão logística com efeitos aleatórios foi utilizado o banco de dados *Surgical*, da própria lista de exemplos do INLA (mathstat.helsinki.fi/openbugs/Examples/Surgical.html) e originalmente apresentado no manual de exemplos do WinBUGS. Tem-se o número de bebês mortos em cirurgias cardíacas de 12 hospitais. O número de cirurgias em cada hospital é diferente, dessa forma podemos definir o número de mortes em cada um dos 12 hospitais como sendo:

$$r_i \sim \text{Binomial}(n_i, p_i)$$

onde, r_i representa o número de mortes no hospital i , p_i a probabilidade de morte em cirurgias cardíacas de bebês no hospital i e n_i o número de cirurgias cardíacas em bebês realizadas no hospital i .

Neste modelo estaremos interessados na probabilidade de morte comum a todos os hospitais e também a uma parte aleatória que se refere as peculiaridades de cada hospital.

Para utilizar o modelo de regressão logística se faz necessário uma transformação via função logito na probabilidade de morte (p_i). Neste caso teremos o seguinte modelo de regressão logística com efeitos aleatórios,

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + v_i$$

onde β_0 é uma constante e representa o logito da probabilidade de morte comum a todos os hospitais e v_i representa o efeito aleatório de cada hospital.

Uma outra forma de ver o mesmo problema seria unindo a constante β_0 com os efeitos aleatórios (padrão nas programações utilizadas no WinBUGS, chamada de centralização hierárquica), desta forma teríamos o seguinte modelo

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_i$$

onde β_i é o logito da probabilidade de morte de cada hospital i .

Para mais detalhes sobre o modelo de regressão logística com efeitos aleatórios veja, por exemplo [7].

Neste modelo os parâmetros desconhecidos são a probabilidade de morte comum a todos os hospitais, o efeito aleatório de cada hospital e também a variância dos efeitos aleatórios. Desta forma temos o vetor $x = \{\beta_0, v_i\}$ e o vetor $\theta = (\sigma^2)$ onde σ^2 é a variância dos efeitos aleatórios. Por definição do método o vetor x deve assumir distribuição Normal, dado o vetor paramétrico θ , ou seja, $\pi(x|\theta)$ deve assumir distribuição Normal. Neste caso o vetor x depende do vetor θ pois temos um efeito aleatório. Portanto atribuiremos distribuição Normal com média 0 e logaritmo da precisão $\theta = \log(1/\sigma^2)$ para v_i e para β_0 atribuiremos distribuição Normal completamente especificada ou seja $\pi(x|\theta)$ será o produto de duas distribuições Normais.

Utilizando este modelo estamos mais interessados na distribuição a posteriori para x , ou seja, estamos interessados em obter $\pi(x|y)$.

Para implementar esse modelo utilizando o *R-INLA*, devemos primeiramente imputar o conjunto de dados no software. O conjunto de dados encontra-se disponível no pacote *R-INLA* e pode ser acessado pelo comando *Surg* (Linha 2 do Quadro (6)).

```
1 library(INLA)
2 data(Surg)

3 formula = r ~ f(hospital,model="iid",param=c(0.001,0.001))
4 resultado = inla(formula,data=Surg,family="binomial",Ntrials=n)

5 summary(resultado)
```

Quadro 6: Programações iniciais para o exemplo de Regressão Logística com Efeitos Aleatórios.

Neste banco temos as seguintes variáveis: *hospital* que é uma identificação para os hospitais, r que indica o número de mortes em cirurgias cardíacas de bebês em cada hospital e n que indica o número de cirurgias cardíacas em bebês em cada hospital.

Em seguida na linha 3 do Quadro (6) definimos a formula, ou seja, a estrutura do modelo. Esta formulação está indicando que o número de mortes por cirurgia cardíacas em bebês (r) pode ser explicado por um efeito aleatório relacionado a cada hospital, também está considerando que

os hospitais são independentes entre si através do comando `model="iid"`, por último a opção `param` está especificando a priori para o logaritmo da precisão de β , já que a média (default na programação) é 0. Interessante notar que nós não informamos ao modelo que queremos informações sobre a constante, isso pelo fato de que a constante vem por padrão no método.

Como priori para o logaritmo da precisão de v_i utiliza-se a distribuição Log-gama. Esta distribuição possui dois parâmetros, que serão denotados por a e b . Neste exemplo utilizou-se $a=b=0.001$, essa escolha foi feita para podermos comparar resultados do método INLA com MCMC (já que esses são os valores utilizados na priori Gama para a precisão no exemplo no manual do WinBUGS). Esta priori foi definida na Linha 3 do Quadro (6).

Na linha 4 do Quadro (6) estaremos aplicando o método INLA para ajustar o modelo de regressão logística com efeitos aleatórios definidos pela *formula*, no banco de dados *Surg*. Estes dados tem distribuição Binomial com parâmetro n conhecido (coluna "n" do banco Surg).

Com o modelo totalmente especificado podemos então acessar seus resultados. Na linha 5 do Quadro (6) acessamos as primeiras informações sobre o modelo especificado. No Quadro (7) tem-se os resultados desse comando.

1	Call:							
2	"inla(formula = formula, family = binomial; data = Surg, Ntrials = n)"							
3	Time used:							
4	Pre-processing	Running inla	Post-processing	Total				
5	1.964	0.816	0.346	3.126				
6	Fixed effects:							
7		mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
8	(Intercept)	-2,5522	0,1483	-2,8606	-2,5477	-2,2699	-2,5393	0
9	Random effects:							
10	Name Model							
11	hospital IID model							
12	Model hyperparameters:							
13		mean	sd	0.025quant	0.5quant	0.975quant	mode	
14	Precision for hospital	9,078	7,668	1,717	6,918	29,246	4,132	
15	Expected number of effective parameters(std dev): 7.887(1.249)							
16	Number of equivalent replicates : 1.522							
17	Marginal Likelihood: -46.14							

Quadro 7: Saída resumo contendo informações básicas do modelo.

No Quadro (7), as duas primeiras linhas estão informando o modelo ajustado. Na linha 3 até a linha 5 temos algumas informações sobre o tempo de processamento deste modelo. Na linha 6 até a linha 8 temos informações sobre os efeitos fixos do modelo (neste caso β_0). Na

linha 9 até a linha 11 temos o nome do efeito aleatório e o tipo de modelo que foi utilizado (neste caso efeito aleatório de hospitais considerando os hospitais independentes entre si). Na linha 12 até a linha 14 temos estatísticas descritivas sobre os hiperparâmetros do modelo (neste caso a precisão do efeito aleatório). Na linha 15 até linha 17 temos informações sobre o número efetivos de parâmetros. Para mais informações sobre o número efetivo de parâmetros ver [3].

O mais importante neste exemplo é obter a distribuição de β_0 que representa o logaritmo da probabilidade de morte comum a todos os hospitais. As informações referentes a este parâmetro podem ser obtida observando as linhas 6, 7 e 8 do Quadro (7) ou então através da função,

```
intercepto = resultado$summary.fixed
```

Essas estatísticas são para o parâmetro β_0 . Podemos ver média, desvio padrão, quantil 0.025, quantil 0.5 ou mediana, quantil 0.975, moda e kld. Este último indica a diferença entre a curva Gaussiana teórica e a Aproximação de Laplace Simplificada (utilizada no método INLA), valores próximos a 0 indicam que a distribuição a posteriori utilizando o método INLA é bem aproximada por uma distribuição Normal, tal que valores altos indicam que se faz necessário o uso da Aproximação de Laplace completa. Ou seja, temos um conjunto de informações a posteriori para o parâmetro β_0 .

Podemos também plotar o gráfico para a distribuição a posteriori marginal de β_0 .

```
plot(resultado, plot.fixed.effects = TRUE, plot.random.effects =  
FALSE, plot.hyperparameters = FALSE)
```

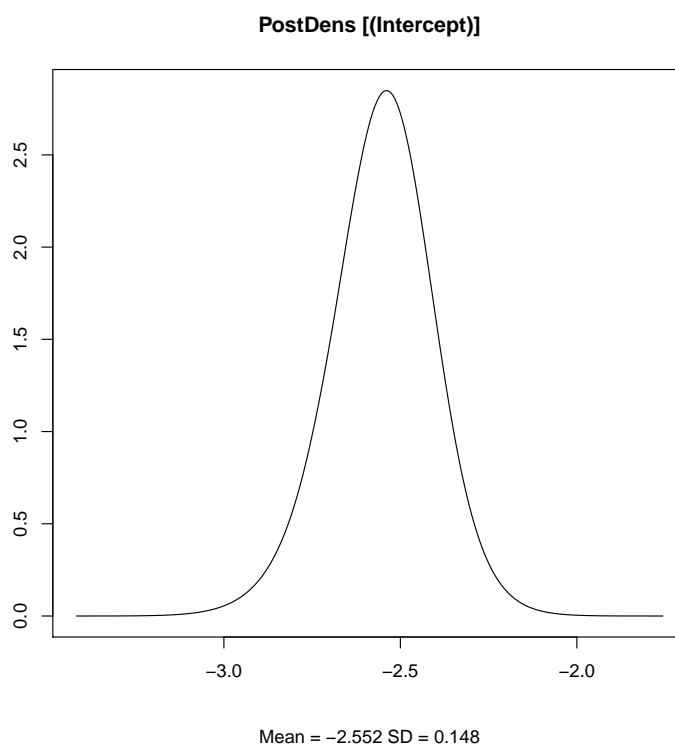


Figura 10: Distribuição marginal a posteriori para o logaritmo da probabilidade de morte de bebês em cirurgias cardíacas.

Uma opção interessante é a de plotar este gráfico já com a transformação $p = \frac{\exp(\beta_0)}{1 + \exp(\beta_0)}$, isso pode ser feito utilizando os comandos das linhas 1, 2 e 3 do Quadro (8),

```

1 marg.beta = resultado$marginals.fixed$(‘Intercept’)
2 marg.p = inla.tmarginal(function(b) exp(b)/(1+exp(b)), marg.beta)
3 plot(marg.p,type=“l”)

4 inla.emarginal(function(x) x),marg.p)
5 inla.mmarginal(marg.p)
6 inla.qmarginal(0.5,marg.p)
7 INLA:::summary.inla.marginal(marg.p)

```

Quadro 8: Estatísticas descritivas para a distribuição a posteriori marginal da probabilidade de morte comum a todos os hospitais de bebês mortos em cirurgias cardíacas.

A Figura (11) apresenta o gráfico para a distribuição a posteriori para o parâmetro p .

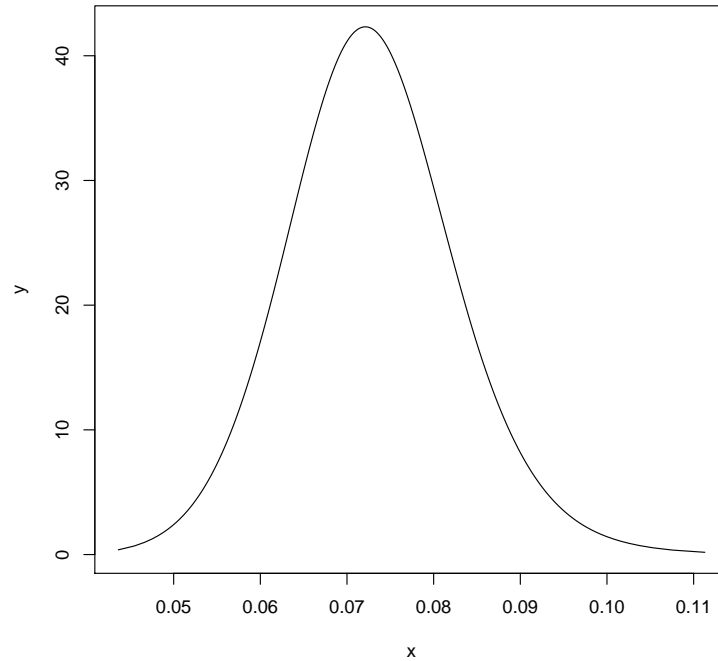


Figura 11: Distribuição marginal a posteriori para a probabilidade de morte de bebês em cirurgias cardíacas.

Para acessar as estatísticas como moda, média, mediana e desvio padrão, já com a transformação, podemos utilizar funções descritas nas linhas 4, 5 e 6 do Quadro (8), ou então de forma mais simples utilizar o comando da linha 7 do Quadro (8) que reúne todas as informações em uma só saída.

Desta forma podemos ver que em média a probabilidade de morte de bebês comum a todos os hospitais é aproximadamente 0.073, ou seja, estima-se que em média 7,3% das crianças que realizam cirurgias cardíacas nestes hospitais venham a óbito. Com 0.95 de probabilidade a verdadeira probabilidade é um número entre 0,054 e 0,093.

Podemos também observar o gráfico dos efeitos aleatórios do modelo, para isso basta utilizar o comando dos gráficos e pedir a opção *plot.random.effects*.

```
plot(resultado, plot.fixed.effects = FALSE, plot.random.effects =  
      TRUE, plot.hyperparameters = FALSE)
```

Podemos ver pelas estatísticas pontuais dispostas na Figura (12) que alguns hospitais tem efeitos aleatórios bem parecidos (caso dos hospitais C, F, G, I, J e L) estes hospitais tem efeitos aleatórios próximos de 0, o que indica que suas estimativas para a probabilidade de morte são bem próximas da probabilidade de morte global estimada.

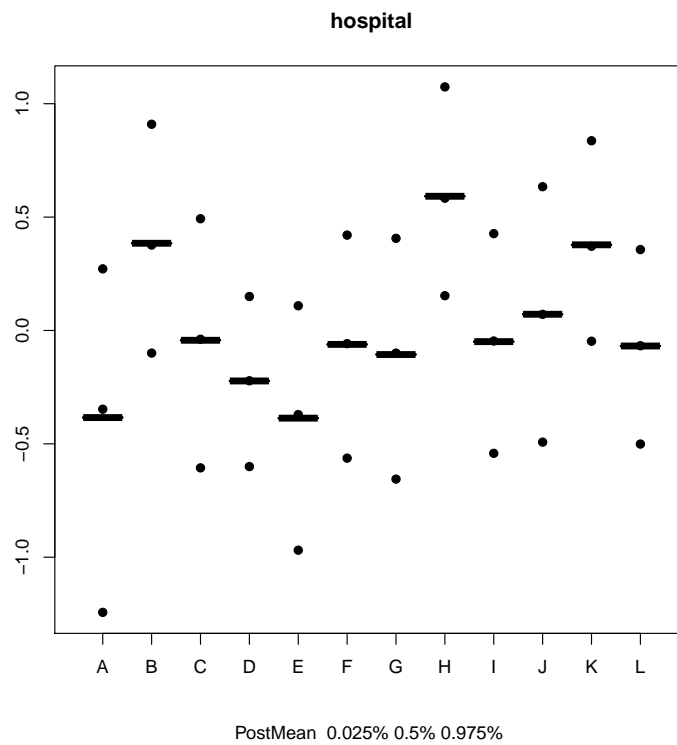


Figura 12: Efeitos aleatórios para os 12 hospitais estudados. Pontos indicam os quantis (0.025, 0.5 e 0.975) e o traço indica a média.

Para acessarmos as informações referentes aos hiperparâmetros do modelo, neste caso a precisão dos efeitos aleatórios de hospitais, podemos utilizar o seguinte comando,

```
precisao = resultado$summary.hyperpar
```

E para plotar o gráfico,

```
plot(resultado, plot.fixed.effects = FALSE, plot.random.effects =
      FALSE, plot.hyperparameters = TRUE)
```

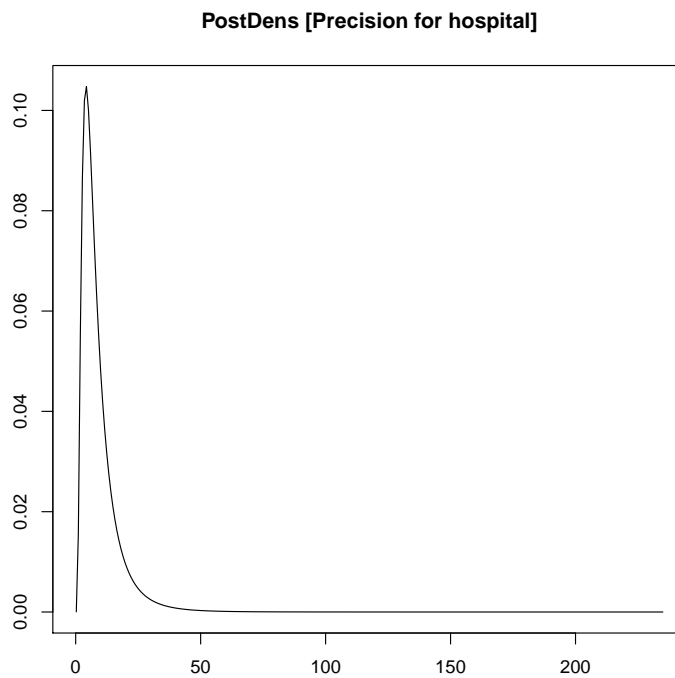


Figura 13: Distribuição marginal a posteriori para a precisão dos efeitos aleatórios do modelo de regressão logística com efeitos aleatórios.

Para obtermos o gráfico da variância podemos utilizar os comandos das linhas 1, 2 e 3 do Quadro (9),

```

1 marg.prec=resultado$marginals.hyperpar$'Precision for hospital'
2 marg.var = inla.tmarginal(function(x) 1/x, marg.prec)
3 plot(marg.var,type="l")

4 inla.emarginal(function(x) x),marg.var)
5 inla.mmarginal(marg.var)
6 inla.qmarginal(0.5,marg.var)
7 INLA:::summary.inla.marginal(marg.var)

```

Quadro 9: Estatísticas descritivas para a distribuição a posteriori marginal da variância dos efeitos aleatórios.

Resultando no seguinte gráfico,

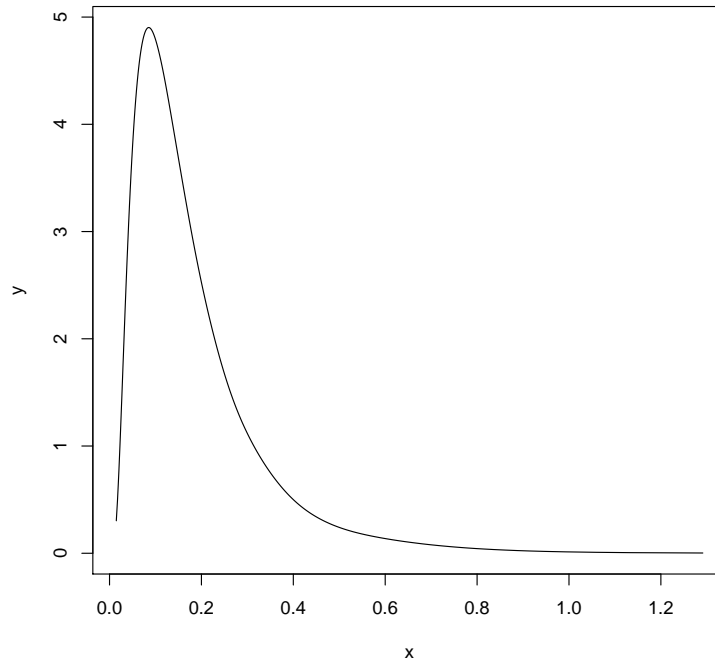


Figura 14: Distribuição marginal a posteriori para a variância dos efeitos aleatórios do modelo de regressão logística com efeitos aleatórios.

As estatísticas para a variância podem ser obtidas utilizando os códigos das linhas 4, 5 e 6, ou então de forma mais fácil utilizando o código da linha 7 do Quadro (9).

3.1.1 Comparação com o WinBUGS.

A fim de compararmos os resultados obtidos pelo método INLA, repetimos as análises no software WinBUGS. Primeiramente definimos a semente aleatória sendo o valor “1”, em seguida utilizamos o modelo já existente no arquivo de exemplos do WinBUGS e utilizamos os mesmos valores iniciais para os parâmetros. Como priori para a precisão optamos pela distribuição Gama, desta forma podemos utilizar uma distribuição análoga no INLA (Log-gama). Foi gerada uma cadeia de 11000 observações e um período de aquecimento de 1000 observações.

No modelo que consta no manual de exemplos, os efeitos aleatórios não estão na mesma formatação que nós implementamos no *R-INLA*. Também não temos estatísticas a respeito da variância dos efeitos aleatórios. Para contornar este problemas duas linhas de código foram adicionadas e obtivemos o seguinte modelo,

```

model
{
for( i in 1 : N ) {
b[i] ~ dnorm(mu,tau)
r[i] ~ dbin(p[i],n[i])
logit(p[i]) <- b[i]
b2[i] <- b[i]-mu
}
pop.mean <- exp(mu) / (1 + exp(mu))
mu ~ dnorm(0.0,1.0E-6)
#Priori
tau ~ dgamma(1.0E-3, 1.0E-3);
sigma <- 1/sqrt(tau); # s.d. of random effects
sigma2 <- sigmasigma
}

```

Os resultados encontram-se na Tabela (1),

Tabela 1: Inferências para os parâmetros do modelo de regressão logística com efeitos aleatórios. INLA vs WinBUGS.

Parâmetro	Estimativas INLA				Estimativas WinBUGS		
	Média	Moda	Mediana	ICr (0,95)	Média	Mediana	ICr (0,95)
\mathbf{p}	0,073	0,072	0,073	(0,054; 0,093)	0,073	0,072	(0,053; 0,094)
σ^2	0,185	0,085	0,144	(0,085; 0,144)	0,188	0,1439	(0,025; 0,620)
v_1	-0,384	-0,279	-0,347	(-1,243; 0,272)	-0,404	-0,344	(-1,370; 0,259)
v_2	0,385	0,358	0,377	(-0,100; 0,910)	0,371	0,357	(-0,099; 0,931)
v_3	-0,043	-0,031	-0,039	(-0,606; 0,493)	-0,053	-0,047	(-0,615; 0,497)
v_4	-0,223	-0,220	-0,222	(-0,600; 0,150)	-0,215	-0,213	(-0,605; 0,164)
v_5	-0,387	-0,339	-0,371	(-0,970; 0,110)	-0,382	-0,362	(-1,003; 0,097)
v_6	-0,061	-0,052	-0,058	(-0,563; 0,421)	-0,066	-0,061	(-0,560; 0,407)
v_7	-0,106	-0,088	-0,100	(-0,655; 0,406)	-0,108	-0,100	(-0,662; 0,417)
v_8	0,592	0,564	0,583	(0,153; 1,074)	0,574	0,564	(0,104; 1,097)
v_9	-0,049	-0,041	-0,046	(-0,542; 0,427)	-0,053	-0,050	(-0,542; 0,433)
v_{10}	0,071	0,069	0,071	(-0,492; 0,634)	0,064	0,059	(-0,488; 0,638)
v_{11}	0,378	0,358	0,371	(-0,047; 0,837)	0,369	0,359	(-0,062; 0,862)
v_{12}	-0,068	-0,065	-0,067	(-0,501; 0,357)	-0,065	-0,064	(-0,494; 0,371)

A coluna ICr(0.95) refere-se ao Intervalo de Credibilidade Central de 0.95 de probabilidade. Para obter este intervalo, por ele ser central, basta encontrar os quantis 0.025 e 0.0975 das distribuições em questão.

Podemos ver que nem as estimativas dos parâmetros nem os ICr's diferem muito entre os dois softwares, e, neste caso, a diferença de tempo entre os dois métodos já foi considerável,

mesmo sendo um exemplo simples (INLA: 3.126s; WinBUGS: 6s).

3.2 Componentes de Variância

Para exemplificar o modelo de componentes de variância foi utilizado o banco de dados Dyes (<http://mathstat.helsinki.fi/openbugs/Examples/Dyes.html>). Este banco apresenta o rendimento de corantes em 6 diferentes lotes, de onde foram retirados 5 observações de corantes. Desta forma podemos definir o modelo como segue,

$$y_{ij} = \mu + B_i + \epsilon_{ij}$$

onde, y_{ij} é o rendimento do corante número j no lote i , μ é a média geral de rendimento dos corantes, B_i é o efeito que o corante (bloco) i tem sobre a observação y_{ij} e ϵ_{ij} é o efeito sobre y_{ij} que não é explicado pelo bloco i , ou seja, as variações naturais dentro dos corantes (erro aleatório). Desta forma assumiremos a priori que $B_i \sim N(0, \sigma_b^2)$ e $\epsilon_{ij} \sim N(0, \sigma_\epsilon^2)$.

O objetivo do estudo é comparar a variação do rendimento dos corantes entre os lotes versus a variação devido a aleatoriedade e então observar a importância relativa dessas variações no modelo. Neste caso as médias destes efeitos não são importantes já que estamos interessados apenas nas variabilidades.

Neste modelo temos como parâmetros desconhecidos a variância do bloco (entre lotes) e a variância do erro aleatório (dentro de lotes) e o intercepto (μ).

De acordo com a notação do modelo hierárquico em (6), neste modelo definimos o vetor $x = \{\mu\}$ e o vetor $\theta = \{\sigma_b^2, \sigma_\epsilon^2\}$, onde, μ é a média geral, σ_b^2 a variância dos blocos e σ_ϵ^2 a variância dos erros aleatórios.

O *R-INLA* utiliza por padrão priori Log-gama para o logaritmo das precisões, ou seja, podemos definir,

$$\begin{aligned} \theta_b &= \log\left(\frac{1}{\sigma_b^2}\right) \\ &\text{e} \\ \theta_\epsilon &= \log\left(\frac{1}{\sigma_\epsilon^2}\right) \end{aligned}$$

Para implementar este modelo no *R-INLA*, temos que primeiramente ler o conjunto de dados (Linhas 1 e 2 do Quadro (9)), que originalmente está em formato matricial, onde as colunas representam as 5 observações e as linhas os 6 blocos.

```

1 library(INLA)
2 dados = data.frame(bloco = rep(1:6, each=5), rendimento = c(1545, 1440, 1440, 1520, 1580,1540, 1555, 1490, 1560, 1495,
1595, 1550, 1605, 1510, 1560, 1445, 1440, 1595, 1465, 1545, 1595, 1630, 1515, 1635, 1625, 1520, 1455, 1450, 1480, 1445))
3 formula = rendimento ~ 1 + f(bloco,model="iid",param=c(0.001,0.001))

4 me = mean(dados$rendimento)
5 dados$rendimento = dados$rendimento - me

6 resultado = inla(formula,data=dados,family="gaussian", control.fixed = list(mean.intercept = -me,
prec.intercept = 1/1e10), control.family = list(hyper = list(prec = list(param = c(0.001, 0.001))))))

```

Quadro 10: Programações iniciais para o exemplo de componentes de variância.

Neste banco temos as seguintes variáveis: *bloco* que é uma identificação para os blocos de corantes e *rendimento* que indica o rendimento dos corantes em cada um dos blocos.

A formulação utilizada na linha 3 está indicando que o rendimento dos corantes é explicado através de uma constante acrescido de um efeito aleatório referente ao bloco. Também está considerando que os blocos são independentes entre si através do comando *model="iid"*. Por último a opção *param* está especificando os parâmetros da distribuição Log-gama, utilizada como priori para o logaritmo da precisão do bloco. A escolha dos parâmetros deu-se para no fim podermos comparar resultados do *R-INLA* com resultados do WinBUGS.

Utilizando os dados na escala original o *R-INLA* retorna erros, pois, este exemplo possui pouca informação (ambas as variâncias são muito grandes, o que confunde o otimizador do método), para contornar este problema faremos uma pequena modificação nos dados. Podemos reescrever o modelo definindo uma variável $z_{ij} = y_{ij} - \hat{\mu}$, onde $\hat{\mu}$ é a média amostral. Esta mudança está sendo realizada nas linhas 4 e 5 do Quadro (10).

$$z_{ij} = \mu_1 + B_i + \epsilon_{ij}$$

Como descontamos a média amostral (e não a verdadeira média já que não a conhecemos) temos ainda que estimar um intercepto. Espera-se que este efeito tenha estimativas próximas a 0. Chamaremos esse novo efeito fixo de μ_1 .

Em seguida na linha 7 definimos onde aplicar o método INLA. Estaremos aplicando o método INLA em um modelo de componentes de variância definidos pela *formula*, no banco de dados *Dados*. Quando especificamos que os dados seguem distribuição Normal em *family* o software já insere o erro aleatório gaussiano no modelo (ϵ). Em *control.fixed* na opção *mean.intercept* estamos dizendo ao método que a priori a média do intercepto (μ_1) é igual ao negativo da média do rendimento (pois subtraímos a média em todas as observações y_{ij}) e informando a precisão

a priori para o intercepto em *prec.intercept*. Em *control.family* estamos indicando a priori para a precisão dos efeitos aleatórios do modelo, ou seja, os valores a priori para $\frac{1}{\sigma_\epsilon^2}$.

Com o modelo totalmente especificado podemos então acessar seus resultados. O primeiro resultado a ser observado e que resume todas as informações do modelo é: *summary(resultado)*

```

1 Call:
2 c("inla(formula = formula, family = gaussian; data = dados, control.family = list(hyper = list(prec = list(param =
  c(0.001, ", "0.001))))), control.fixed = list(mean.intercept = -me, prec.intercept = 1/1e+10)")

3 Time used:
4 Pre-processing          Running inla  Post-processing  Total
5 0.984                  0.203         0.0469          1.2344

6 Fixed effects:
7                   mean          sd          0.025quant  0.5quant  0.975quant  mode      kld
8 (Intercept)       -0.003       22.539        -46.2773    -0.0033    46.307      -1.00E-04  0

9 Random effects:
10 Name              Model
11 bloco             IID          model

12 Model hyperparameters:
13                   mean          sd          0.025quant  0.5quant  0.975quant  mode
14 Precision for the Gaussian observations 0.0004    0.0001    0.0002     0.0004    0.0007     0.0004
15 Precisionfor bloco 0.0009    0.001     0.0001     0.0006    0.0036     0.0003

16 Expected number of effective parameters (stddev): 4.679(0.8653)
17 Number of equivalent replicates:6.412

18 Marginal Likelihood: -185.64

```

Quadro 11: Saída resumo contendo informações básicas do modelo.

No Quadro (11) as duas primeiras linhas estão informando o modelo ajustado. Na linha 3 até a linha 5 temos informações sobre o tempo de processamento deste modelo. Na linha 6 até a linha 8 temos informações sobre o único efeito fixo do modelo (média geral do rendimento). Na linha 9 até a linha 11 temos informações sobre os efeitos aleatórios do modelo (neste caso os blocos). Na linha 12 até a linha 15 temos informações sobre os hiperparâmetros do modelo (neste caso a precisão do efeito aleatório e bloco e a precisão do efeito aleatório do erro aleatório). Na linha 16 até linha 18 temos informações sobre o número efetivos de parâmetros. Para mais informações sobre o número efetivo de parâmetros ver [3].

O mais importante neste exemplo é obter a distribuição de θ_b e de θ_ϵ com o intuito de observar se a variação entre blocos é importante para o modelo. As informações referentes a estes parâmetros podem ser vistas nas linhas 12, 13, 14 e 15 ou então através da seguinte função,

```
resultado$summary.hiperpar
```

Podemos plotar os gráficos da precisão dos efeitos aleatórios a posteriori com o seguinte comando.

```
plot(resultado, plot.fixed.effects = FALSE, plot.random.effects =
      FALSE, plot.hyperparameters = TRUE)
```

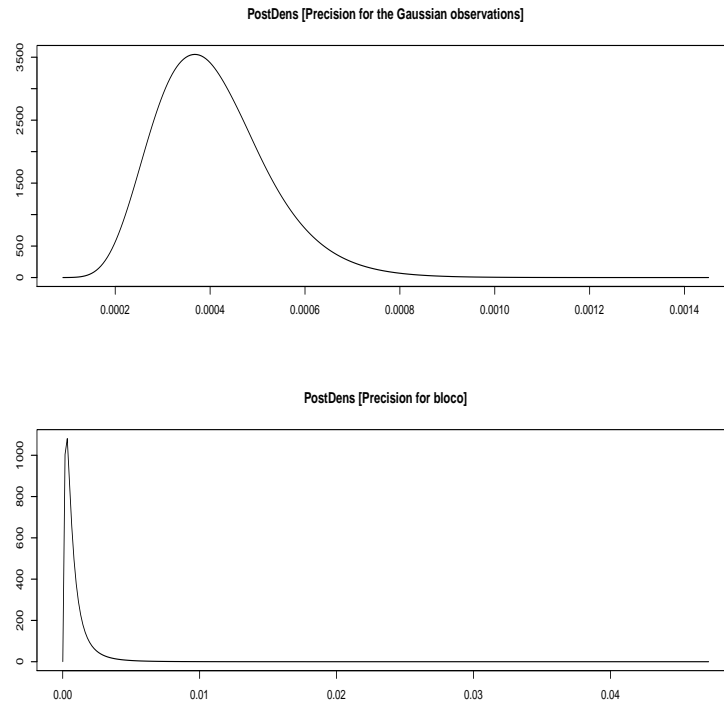


Figura 15: Distribuição marginal a posteriori para a precisão dos efeitos aleatórios do modelo de componentes de variância.

Porém o mais interessante é fazer a transformação da posteriori a fim de obtermos a distribuição a posteriori para a variância, da seguinte forma,

```
marg.var.bloco=inla.tmarginal(function(x) 1/x,marg.bloco)
marg.var.ea=inla.tmarginal(function(x) 1/x,marg.ea)
par(mfrow=c(2,1))
plot(inla.smarginal(marg.var.bloco),type='l')
plot(inla.smarginal(marg.var.ea),type='l')
```

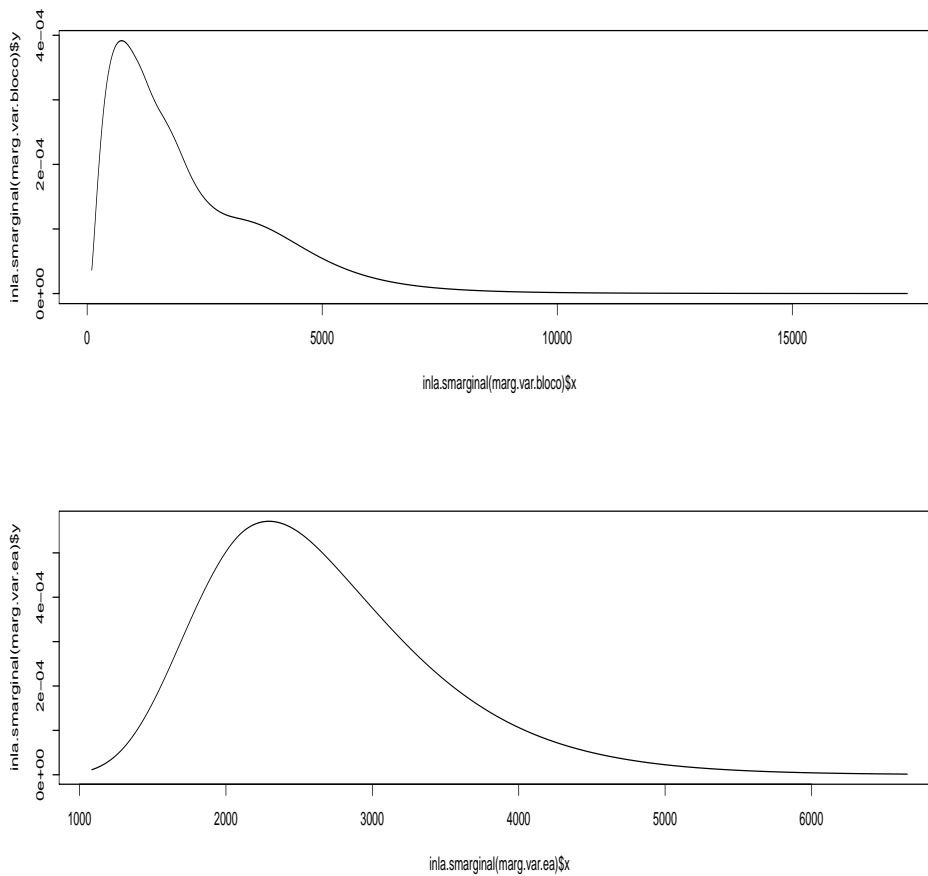


Figura 16: Distribuição marginal a posteriori para a variância dos efeitos aleatórios do modelo de componentes de variância.

Vê-se que ambas as variâncias são muito grandes, porém aparentemente a variância do bloco é menor pois está mais concentrada em valores menores.

3.2.1 Comparação com o WinBUGS

A fim de compararmos os resultados obtidos pelo método INLA, repetimos as análises no software WinBUGS. Primeiramente definimos a semente aleatória sendo o valor “1”, em seguida utilizamos o modelo já existente no arquivo de exemplos do WinBUGS e utilizamos os mesmos valores iniciais para os parâmetros. Como priori para a precisão optamos pela distribuição Gama, desta forma podemos utilizar uma distribuição análoga no INLA (Log-gama). Foi gerada uma cadeia de 100000 observações e um período de aquecimento de 25000 observações.

Os resultados encontram-se na Tabela (2),

Tabela 2: Inferências para os parâmetros do modelo de componentes de variância. INLA vs WinBUGS.

Parâmetro	Estimativas INLA				Estimativas WinBUGS		
	Média	Moda	Mediana	ICr (0,95)	Média	Mediana	ICr (0,95)
μ	1.527,50	1.527,51	1.527,50	(1481,271; 1573;769)	1.527,00	1.527,00	(1485; 1571)
σ_b^2	2.201,30	731,55	1.659,42	(896,733 ; 6508,130)	2.194,00	1.277,00	(0.006; 10230)
σ_e^2	2.675,89	2.294,24	2.541,42	(1476,76; 4613,71)	3.059,00	2.815,00	(1562; 5880)

Para obtermos as estatísticas para μ pelo *R-INLA*, primeiramente encontramos estatísticas para μ_1 e em seguida somamos o valor estimado para a média no modelo original.

Podemos ver que para este caso tivemos grandes diferenças entre os dois métodos para as variâncias. O WinBUGS coloca uma observação sobre a priori Gama utilizada não aconselhando o uso da mesma, porém utilizamos esta priori justamente para podermos comparar resultados.

Este exemplo tem pouca informação, visto que ambas as variâncias são muito grandes, desta forma temos intervalos de credibilidades que não nos informam muito já que são muito amplos. Os intervalos do INLA no entanto foram menos amplos que os obtidos por MCMC. Em estatística clássica este modelo teria componente de variância negativo, o que não faz sentido.

Outro aspecto interessante é que o *R-INLA* já nos traz automaticamente a moda da distribuição, já no WinBUGS seria necessário calcular este valor através de estimação não-paramétrica da densidade da posteriori.

Neste exemplo tivemos uma grande diferença no tempo computacional entre os métodos (INLA: 1,2344s; WinBUGS: 62s). Isso deve-se ao fato de termos utilizado uma cadeia bastante longa neste exemplo, porém isso foi necessário para que os resultados fossem mais confiáveis.

3.3 Medidas Repetidas

Para exemplificar o modelo de medidas repetidas foi utilizado o banco de dados Rats (<http://mathstat.helsinki.fi/openbugs/Examples/BiRats.html>). Este banco apresenta o peso de 30 ratos ao longo de 5 medições (dias 8, 15, 22, 29 e 30). Desta forma podemos definir o modelo da seguinte forma,

$$y_{ij} = (\alpha + \alpha_i) + (\beta + \beta_i)T_j + \epsilon_{ij}$$

Onde, y_{ij} é o peso do rato i na medição j , α é o peso médio dos ratos ao nascer (intercepto médio), α_i é a diferença aleatória entre o intercepto médio e o intercepto referente ao rato i , β é o coeficiente angular médio de todos os ratos, β_i é a diferença aleatória entre o coeficiente angular médio e o coeficiente angular referente ao rato i , T_j se refere a medição j e ϵ_{ij} é o efeito sobre y_{ij} que não é explicado pelo modelo.

Neste ponto podemos pensar em duas formas de utilizar este exemplo, a primeira seria atribuindo prioris independentes para α_i e β_i , porém é mais interessante pensar no caso onde α_i e β_i são correlacionados. Por exemplo, caso a correlação seja positiva, poderíamos pensar que ratos mais pesados inicialmente terão uma evolução no peso mais acentuada.

O objetivo do estudo é identificar se os ratos tem um comportamento semelhante e se o tempo influencia no aumento de peso dos ratos. Ao mesmo tempo os parâmetros α_i e β_i serão avaliados a fim de checar se realmente há correlação entre eles.

Neste modelo, teremos o vetor $x = \{\alpha, \beta, \alpha_i, \beta_i\}$ e o vetor $\theta = \{\sigma_{\alpha_i}^2, \sigma_{\beta_i}^2, \sigma_{\epsilon}^2, \rho\}$.

Como já sabemos devemos atribuir priori Normal para o vetor x , porém neste exemplo dois elementos deste vetor (α_i e β_i) dependem do vetor θ e além disso são correlacionados. Os outros dois parâmetros (α e β) não dependem do vetor θ e são independentes. Desta forma teremos o produto de três distribuições Normais. Duas delas univariadas e totalmente especificadas (para α e para β) e para o vetor $\{\alpha_i, \beta_i\}$ uma Normal Bivariada com matriz de variâncias e covariâncias denotada por W :

$$W = \begin{pmatrix} \sigma_{\alpha_i}^2 & \rho\sqrt{\sigma_{\alpha_i}^2\sigma_{\beta_i}^2} \\ \rho\sqrt{\sigma_{\alpha_i}^2\sigma_{\beta_i}^2} & \sigma_{\beta_i}^2 \end{pmatrix}$$

Para W utilizaremos priori Wishart.

Ao implementar esse modelo utilizando o *R-INLA*, devemos primeiramente imputar o conjunto de dados no software, o que está nas linhas 1 e 2 do Quadro (12).

```

1 dados = data.frame(rato = rep(1:30, each=5), T = rep(c(8.0, 15.0, 22.0, 29.0, 36.0), 30),
2 Y = matrix(c(151, 199, 246, 283, 320,
145, 199, 249, 293, 354, 147, 214, 263, 312, 328, 155, 200, 237, 272, 297, 135, 188, 230, 280, 323, 159, 210, 252, 298, 331,
141, 189, 231, 275, 305, 159, 201, 248, 297, 338, 177, 236, 285, 350, 376, 134, 182, 220, 260, 296, 160, 208, 261, 313, 352,
143, 188, 220, 273, 314, 154, 200, 244, 289, 325, 171, 221, 270, 326, 358, 163, 216, 242, 281, 312, 160, 207, 248, 288, 324,
142, 187, 234, 280, 316, 156, 203, 243, 283, 317, 157, 212, 259, 307, 336, 152, 203, 246, 286, 321, 154, 205, 253, 298, 334,
139, 190, 225, 267, 302, 146, 191, 229, 272, 302, 157, 211, 250, 285, 323, 132, 185, 237, 286, 331, 160, 207, 257, 303, 345,
169, 216, 261, 295, 333, 157, 205, 248, 289, 316, 137, 180, 219, 258, 291, 153, 200, 244, 286, 324), 30, 5 ,byrow=T) )

3 Nrats = 30
4 dados$Trato = dados$rato + Nrats
5 formula = y ~ 1 + T +
+ (rato, model = "iid2d", n=2*Nrats, hyper=list(theta1=list(param=c(4, 200, 0.2, 0)))) +
+ f(Trato, T, copy="id")
6 resultado = inla(formula, data = dados, family="gaussian")
7 summary(resultado)

```

Quadro 12: Programações iniciais para o exemplo de Medidas Repetidas.

Neste banco temos as seguintes variáveis: *rato* que identifica cada um dos 30 ratos, *T* que identifica a medição e *y* que refere-se ao peso dos ratos.

Nas linhas 3 e 4 do Quadro (12) estamos criando uma nova coluna neste banco de dados, a qual chamaremos de *Trato* que será igual a coluna *rato* somada ao número de ratos. Isto será útil para incorporar correlação entre α_i e β_i .

Na linha 5 do Quadro (12) definimos a estrutura do modelo através da *formula*. Estamos informando que o peso dos ratos ao nascer tem uma média geral (α) e também um efeito fixo “T” referente as 5 medições (β). A primeira função está indicando que existe um efeito aleatório relacionado aos ratos (α_i). Em *model = iid2d* informamos que na verdade teremos dois efeitos aleatórios relacionados ao mesmo rato, em *n* estamos informando o número de ratos, note que como queremos incorporar dois efeitos aleatórios, devemos multiplicar o número de ratos por 2, em *hyper* estamos definindo os valores para serem utilizados na priori para W (mesmos valores adotados no exemplo do WinBUGS). Na segunda função informamos ao método qual será o segundo efeito aleatório relacionado ao rato (*Trato* e *x*), ou seja, para cada rato teremos um efeito relacionado com os tempos das medições (β_i).

Já na linha 6 do Quadro (12) informamos ao método o modelo o qual queremos informações (*formula*), em que variável está o banco de dados e a distribuição dos dados.

Com o modelo totalmente especificado podemos então acessar seus resultados. (Linha 7 do Quadro (12))

1	Call:								
2	inla(formula=formula, family=\ gaussian\ “,data=dados)”								
3	Time used:								
4	Pre-processing	Running inla	Post-processing	Total					
5	0,65	1,31	0,12	2,07					
6	Fixed effects:								
7		mean	sd	0.025quant	0.5quant	0.975quant	mode	kld	
8	(Intercept)	106,5681	2,1370	102,3572	106,5681	110,7739	106,5681	0,0000	
9	T	6,1856	0,1023	5,9840	6,1856	6,3870	6,1857	0,0000	
10	Random effects:								
11	Name	Model							
12	rato	IID2D model							
13	Trato	Copy							
14	Model hyperparameters:								
15		mean	sd	0.025quant	0.5quant	0.975quant	mode		
16	Precision for the Gaussian observations	0,0307	0,0037	0,0237	0,0306	0,0383	0,0306		
17	Precision for rato (component 1)	0,0124	0,0035	0,0067	0,0120	0,0205	0,0112		
18	Precision for rato (component 2)	4,1186	1,2518	2,1316	3,9684	6,9995	3,6911		
19	Rho1:2 for rato	-0,2684	0,1190	-0,4938	-0,2706	-0,0286	-0,2730		
20	Expected number of effective parameters (std dev):	47.58(2,353)							
21	Number of equivalent replicates/;	3.153,00							
22	Marginal Likelihood:	-1140.23							

Quadro 13: Saída resumo contendo informações básicas do modelo.

No Quadro (13), as duas primeiras linhas estão informando o modelo ajustado. Na linha 3 até a linha 5 do Quadro (13) temos informações sobre o tempo de processamento deste modelo. Na linha 6 até a linha 9 do Quadro (13) temos informações sobre os dois

efeitos fixos do modelo (média geral de peso (α) e coeficiente angular médio (β)). Na linha 10 até a linha 13 do Quadro (13) temos informações sobre o modelo utilizado para os efeitos aleatórios e os nomes das variáveis utilizadas. Na linha 14 até a linha 19 do Quadro (13) temos informações sobre os hiperparâmetros do modelo. Nas linhas 20 e 21 do Quadro (13) temos informações sobre o número efetivos de parâmetros. Ver [3].

Para este modelo podemos buscar informações sobre um grande conjunto de parâmetros $(\alpha, \alpha_i, \beta, \beta_i, \sigma_{\alpha_i}^2, \sigma_{\beta_i}^2, \sigma^2, \rho)$, este último aparece pois ao não considerarmos α_i e β_i independentes então acrescentamos a medida de correlação entre esses parâmetros.

Primeiramente observaremos os resultados referentes ao hiperparâmetros do modelo nas linhas 14 a 19 do Quadro (13), ou então pelo comando,

```
resultado$summary.hyperpar
```

Podemos plotar os gráficos da precisão dos efeitos aleatórios a posteriori com os seguintes comandos.

```
par(mfrow=c(2,2))
plot(resultado$marginals.hyperpar$`Precision for the Gaussian
observations`,type=`l`,main=``Precisão das observações``)
plot(resultado$marginals.hyperpar$`Precision for rato (component
1)`,type=`l`,main=``Precisão da componente 1``)
(resultado$marginals.hyperpar$`Precision for rato (component
2)`,type=`l`,main=``Precisão da componente 2``)
plot(resultado$marginals.hyperpar$`Rho1:2 for
rato`,type=`l`,main=``Correlação entre componente 1 e componente 2``)
```

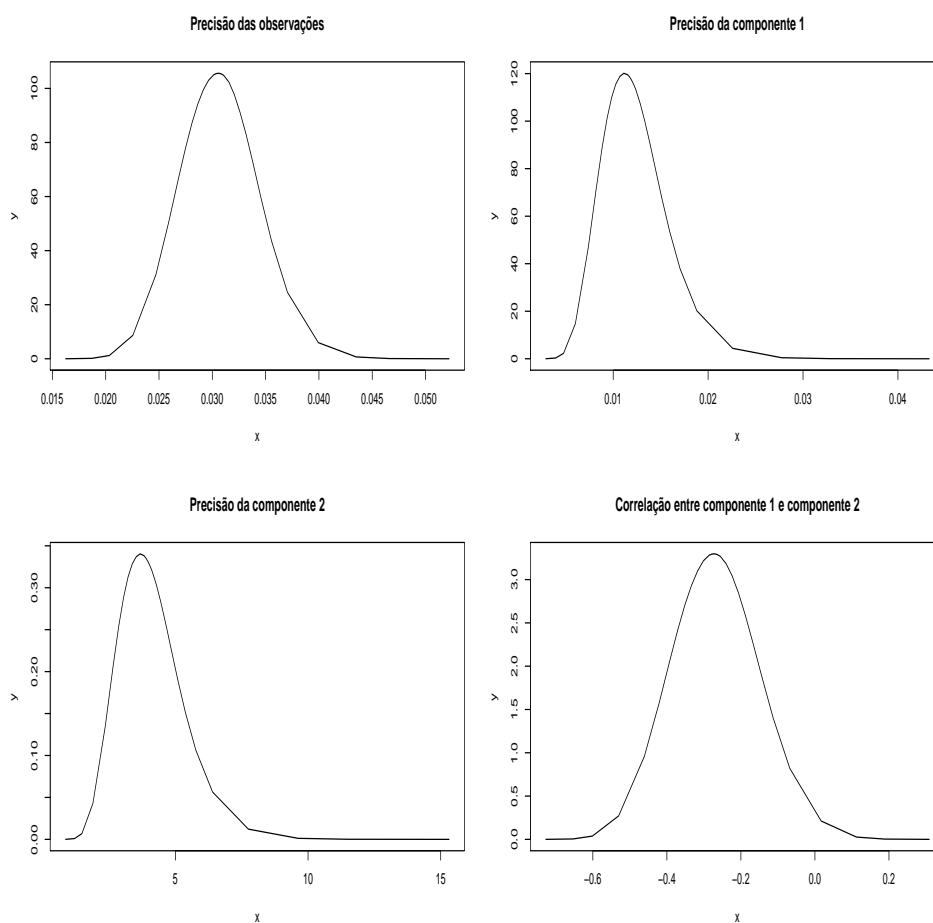


Figura 17: Distribuição marginal a posteriori para a precisão dos efeitos aleatórios e para correlação do modelo de medidas repetidas.

Ou então através do comando:

```
plot(resultado, plot.fixed.effects = FALSE, plot.random.effects
      = FALSE, plot.hyperparameters = TRUE)
```

Porém o mais interessante é fazer a transformação da posteriori a fim de obtermos a distribuição a posteriori para as variâncias, da seguinte forma,

```
marg.prec.gauss=resultado$marginals.hyperpar$`Precision for the Gaussian
                                observations`
marg.prec.c1=resultado$marginals.hyperpar$`Precision for rato (component
                                1)`
marg.prec.c2=resultado$marginals.hyperpar$`Precision for rato (component
                                2)`
```

```

marg.var.gauss=inla.tmarginal(function(x) 1/x,marg.prec.gauss)
marg.var.c1=inla.tmarginal(function(x) 1/x,marg.prec.c1)
marg.var.c2=inla.tmarginal(function(x) 1/x,marg.prec.c2)

par(mfrow=c(3,1))

plot(inla.smarginal(marg.var.gauss),type='l',main='`Variância das
observações`')

plot(inla.smarginal(marg.var.c1),type='l',main='`Variância da componente
1`')

plot(inla.smarginal(marg.var.c2),type='l',main='`Variância da componente
2`')

```

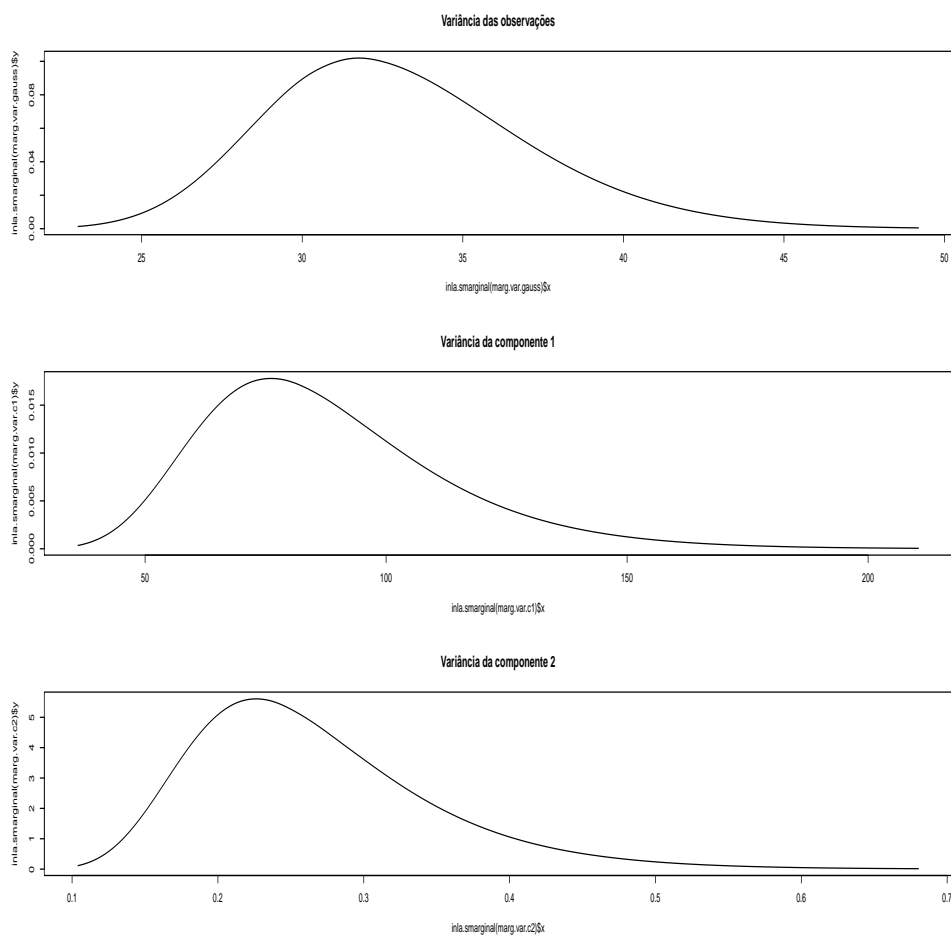


Figura 18: Distribuição a posteriori marginal para a variância dos efeitos aleatórios do modelo de medidas repetidas.

Em todas as saídas o componente 1 está se referindo a constante de cada rato, ou seja α_i e a componente 2 refere-se ao coeficiente angular de cada rato β_i .

Agora veremos informações e gráficos dos dois efeitos fixos do modelo,

```
resultado$summary.fixed
```

	mean	sd	0.025quant	0.5quant	0.975quant	mode	kld
(Intercept)	106.56814	2.1370037	102.357229	106.56807	110.773918	106.568115	1.58E-09
T	6.18565	0.1022741	5.984026	6.185648	6.387007	6.185653	4.58E-09

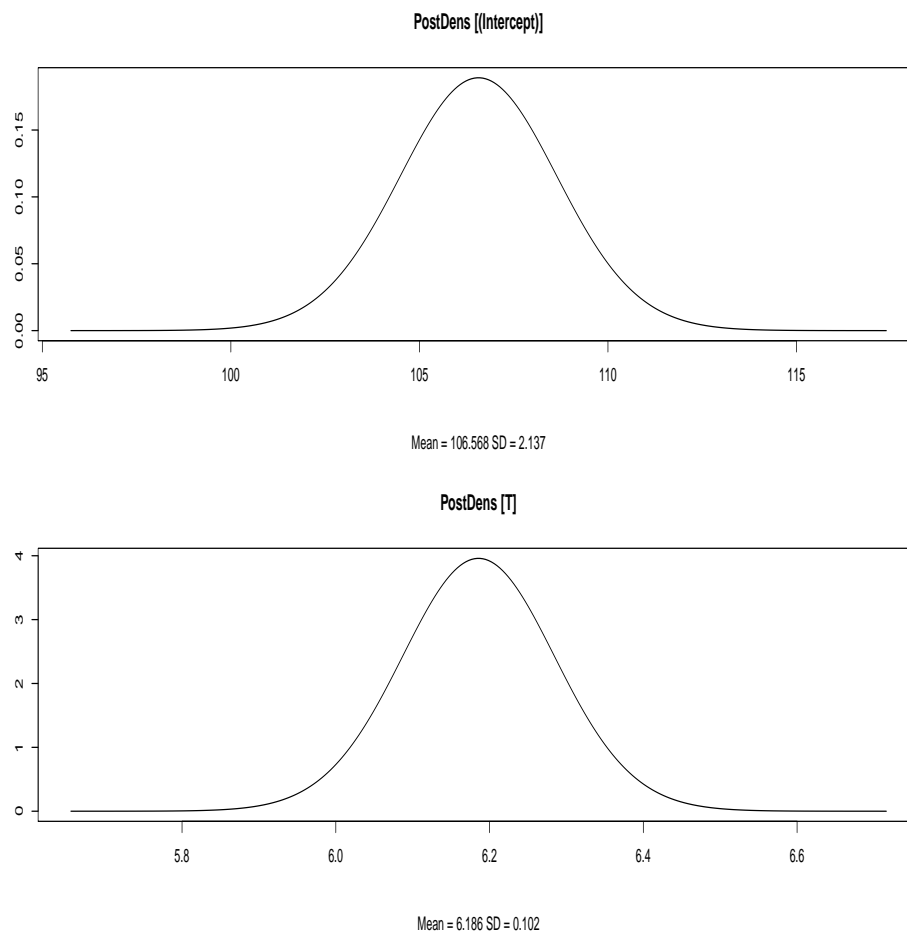


Figura 19: Distribuição marginal a posteriori par os efeitos fixos do modelo de medidas repetidas.

Podemos ver no Quadro (13) que a estimativa da correlação entre α_i e β_i é negativa de forma importante, pois o Intervalo de Credibilidade Central não contém o valor 0 ($\text{ICr}(0.95, \rho) = (-0,4938; -0,0286)$), ou seja, os resultados estão nos dizendo que em geral ratos com um peso inicial maior tendem a ter seu crescimento no peso menor que ratos que nascem menos pesados. Vê-se também que em média a cada dia os ratos aumentam 6,18 gramas em seu peso e que o efeito dos dias é importante, já que o seu intervalo de

credibilidade não contém o valor 0 ($ICr(0.95, \beta) = (5,9840; 6,3870)$). O valor estimado para α foi de aproximadamente 106,57, esse valor representa o intercepto médio, ou seja, o peso médio dos ratos ao nascer.

Podemos ver na Figura (18) que a variância de α_i é maior que a variância de β_i , o que nos indica que em geral os ratos tem pesos mais heterogêneos ao nascer porém a evolução de pesos é mais homogênea.

3.3.1 Comparação com o WinBUGS

A fim de compararmos os resultados obtidos pelo método INLA, repetimos as análises no software WinBUGS. Primeiramente definimos a semente aleatória sendo o valor “1”. Em seguida utilizamos o modelo já existente no arquivo de exemplos do WinBUGS e utilizamos os mesmos valores iniciais para os parâmetros. Foi gerada uma cadeia de 11000 observações e um período de aquecimento de 1000 observações. Os resultados encontram-se na Tabela (3).

Tabela 3: Inferências para os parâmetros do modelo de medidas repetidas. INLA vs WinBUGS

Parâmetro	Estimativas INLA				Estimativas WinBUGS		
	Média	Moda	Mediana	ICr (0,95)	Média	Mediana	ICr (0,95)
α	106.568	106.568	106.568	(102.357; 110.774)	106.600	106.600	(102.000; 111.200)
β	6.186	6.186	6.186	(5.984; 6.387)	6.183	6.183	(5.969; 6.397)
$\frac{1}{\sigma^2}$	0.031	0.031	0.031	(0.024; 0.038)	0.027	0.027	(0.019; 0.036)
$\frac{1}{\sigma_{\alpha_i}^2}$	0.012	0.011	0.012	(0.007; 0.021)	0.010	0.009	(0.005; 0.020)
$\frac{1}{\sigma_{\beta_i}^2}$	4.119	3.691	3.968	(2.132; 7.000)	4.655	4.329	(2.273; 8.830)
ρ	-0.268	-0.273	-0.271	(-0.494; -0.029)	0.0829	0.098	(-0.430; 0.515)

Pode-se notar na Tabela (3) que não houveram grandes divergências entre os resultados trazidos pelos métodos exceto o resultado encontrado para ρ . Pelo método INLA concluiríamos que existe correlação negativa importante entre α_i e β_i ; já nos resultados trazidos pelo WinBUGS não temos este mesmo raciocínio visto que o intervalo de credibilidade é bastante amplo e o valor 0 está contido no intervalo.

Para os efeitos α_i e β_i traçamos retas para 5 ratos escolhidos aleatoriamente e pudemos observar que as retas são praticamente idênticas, ou seja, as estimativas destes parâmetros também estão similares entre os dois métodos, isso pode ser visto na Figura (20).

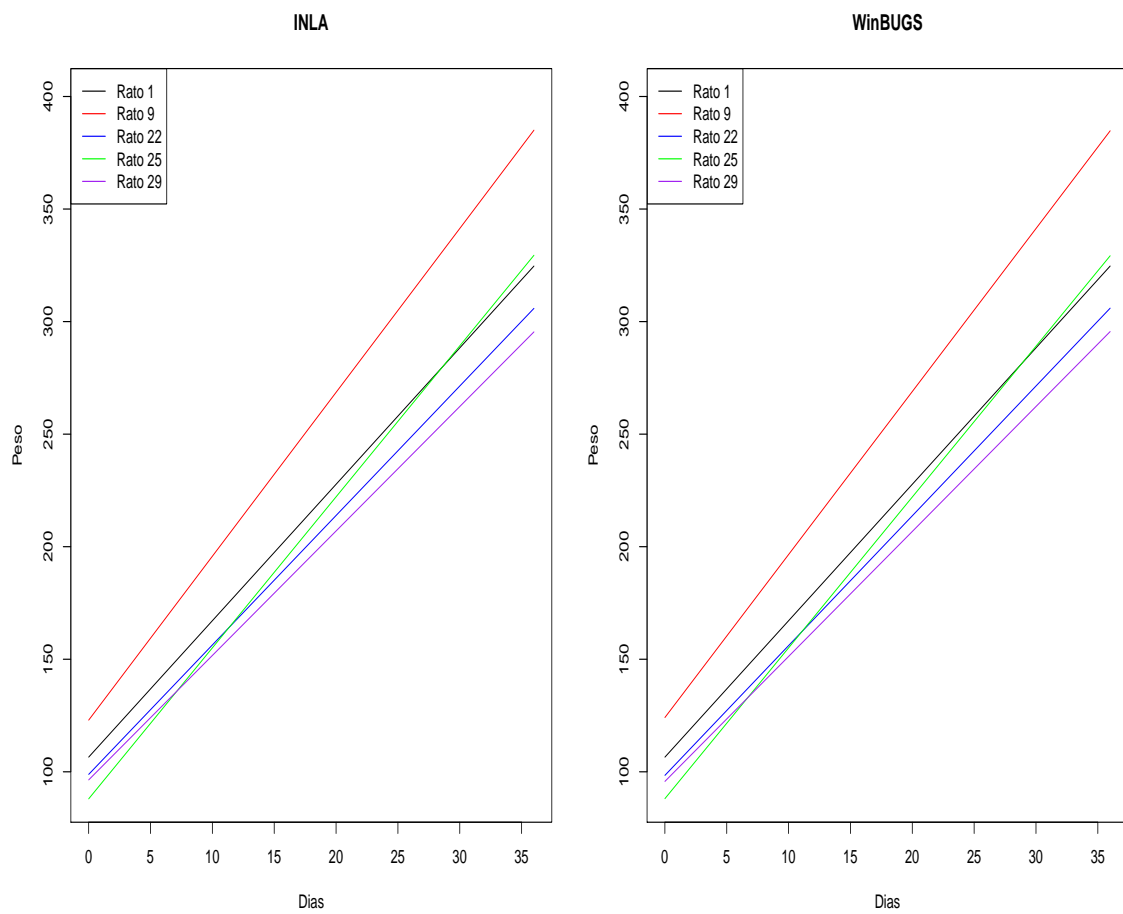


Figura 20: Comparação dos efeitos aleatórios do Modelo de Medidas Repetidas estimados por INLA e WinBUGS.

Desta forma podemos observar que as únicas diferenças importantes entre as duas metodologias (INLA e MCMC) se deu na correlação, neste exemplo.

Também tivemos uma diferença grande nos tempos de processamento (INLA: 2.07s; WinBUGS: 6s).

4 Conclusões

Ao longo deste trabalho enfrentou-se primeiramente uma grande dificuldade em entender o que o método realmente fazia, pois, por se tratar de um método que faz diferença principalmente em modelos complexos (onde o tempo computacional gasto para realização utilizando MCMC não é vantajoso) não haviam referências que explicassem passo a passo a obtenção das distribuições de interesse, e sim referências exemplificando a implementação, no software, de modelos bastante complexos. Então num primeiro momento um grande tempo foi gasto para trazer a matemática deste método para modelos simples onde a matemática ficasse mais clara.

Pode-se observar que por trás do método INLA existe uma matemática um pouco complicada e que sua implementação necessita bastante atenção.

Através de dois exemplos conseguimos mostrar a matemática inserida no método e vimos que mesmo utilizando a aproximação Gaussiana (e não a aproximação de Laplace Simplificada, default no pacote “INLA”), para os simples exemplos propostos, o resultado foi bastante satisfatório.

A implementação utilizando o pacote “INLA” é bastante simples e isto pode ser visualizado através de dois exemplos práticos. Pudemos observar que as funções disponíveis dentro do pacote são bastante uteis e que o método atende uma grande variedade de modelos, além de possuir comandos para a parte gráfica muito eficientes e simples.

Pudemos observar que o *R-INLA* pode apresentar erros na otimização quando utilizado o windows 32 bits e uma possível solução é subtrair de cada observação a média amostral, ou então, alteração nos valores utilizados a priori. Viu-se também que a ajuda online do INLA (help@r-inla.org) é bastante rápida e eficiente.

Notou-se algumas dissimilaridades importantes entre os resultados do *INLA* e do WinBUGS (por exemplo a correlação entre α_i e β_i no exemplo de medidas repetidas). Seria interessante realizar algumas simulações para identificar potenciais fragilidades do pacote *R-INLA*.

Por fim pudemos observar que mesmo em modelos bastante simples, como os abordados neste trabalho, o método INLA tem um tempo computacional bem inferior ao tempo computacional gasto em métodos de simulação.

Um trabalho interessante seria dar uma continuidade a este estudo no sentido de acrescentar a matemática de um modelo com efeitos aleatórios, o que dificultaria a matemática do do método e também a sua programação.

5 Referências

Referências

- [1] Bonat, W. H., Ribeiro Jr, P. J., Aplicações de inferência Bayesiana aproximada em modelos gaussianos latentes espaço-temporais, XIX Simpósio Nacional de Probabilidade e Estatística, vol 1, 1–24, 2010.
- [2] Geman, Stuart and Geman, Donald, Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images, IEEE Trans. Pattern Anal. Mach. Intell., vol 6, 721–741, 1984.
- [3] Baio, Gianluca, An introduction to INLA with a comparison to JAGS (hands on), BAYES 2013 workhop in Rotterdam 21-23 May, 2013.
- [4] Fong, Y., Rue, H. Wakefield, J., Bayesian inference for generalized linear mixed models, Biostatistics., vol 11, 397–412, 2010.
- [5] Gilks, W.R., Richardson S., Spiegelhalter, D.. Markov Chain Monte Carlo in Practice. Chapman and Hall/CRC. 1995.
- [6] James, B. J. Probabilidade: Um Curso Em Nível Intermediário. Editora IMPA. 2009.
- [7] Larsen, Klaus and Petersen, Jørgen Holm and Budtz-Jørgensen, Esben and Endahl, Lars, Interpreting Parameters in the Logistic Regression Model with Random Effects, Biometrics, vol 56, 909–914, 2000.
- [8] Martins, T. G., Aproximações Determinísticas Para Distribuições a Posteriori Marginais, 2010, 98f, Dissertação(Mestrado em Estatística) - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2001.
- [9] Rue, H., Martino, S. and Chopin, N., Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations, Journal of the Royal Statistical Society, vol 71, 319—392, 2009.
- [10] Tierney, Luke and Kadane, Joseph B., Accurate Approximations for Posterior Moments and Marginal Densities, Journal of the American Statistical Association, vol 81, 82–86, 1986.

6 Anexos

Funções utilizada no modelo Normal. (Seção (2.4.2))

$$\pi(\theta)$$

```
pi.gama = function(lp,a,b)
{
  (exp(lp)**a)*(exp(-b*exp(lp)))*(b**a)/gamma(a)
}
```

$$L(y; x, \theta)$$

```
vero = function(lp,x,y)
{
  n=length(y)
  exp(-(exp(lp)/2)*(sum(y**2)-2*x*sum(y)+ n*(x**2)))*((exp(lp)/(2*pi))**(n/2))
}
```

$$\pi(x|\theta)$$

```
pi.x_gama=function(x,lp.mi)
{
  (1/sqrt(2*pi))*exp(lp.mi)*exp(-exp(lp.mi)*(x**2)/2)
}
```

$$\pi_G(x|\theta, y)$$

```
pi.g.x_gamay=function(lp.mi,lp,xm,x,y)
{
  n=length(y)
  aux=sum(y**2) - 2*xm*sum(y) + n*(xm**2)
  fx = n*lp/2 + lp.mi/2 -(exp(lp)/2)*aux - exp(lp.mi)*(xm**2)/2
  n*lp/2 lp.mi/2
}
```

```

flx = exp(lp)*(sum(y)-n*(xm)) - exp(lp.mi)*xm
flx = -n*exp(lp) - exp(lp.mi)
bx0 = flx-flx*xm
cx0 = - flx
dnorm(x, mean=bx0/cx0, sd=sqrt(1/cx0))
}

```

$$\pi(\theta|y)$$

```

pi.gama.post = function(lp,a,b,xm,y,lp.mi)
{
a1=pi.gama(lp=lp,a=a,b=b)
a2=pi.x_gama(x=xm,lp.mi=lp.mi)
a3=vero(lp=lp,x=xm,y=y)
a4=pig.x_gamay(lp.mi=lp.mi,lp=lp,xm=xm,x=xm,y=y)
a1*a2*a3/a4
}

```