



**ALGORITMOS PARA MODELOS DE LIMIAR
USANDO AS DISTRIBUIÇÕES ACUMULADAS
NORMAL E t DE STUDENT**

JOSÉ WALDEMAR DA SILVA

2008

JOSÉ WALDEMAR DA SILVA

**ALGORITMOS PARA MODELOS DE LIMIAR USANDO
AS DISTRIBUIÇÕES ACUMULADAS NORMAL E t DE
STUDENT**

Tese apresentada à Universidade Federal de Lavras
como parte das exigências do Programa de Pós-
graduação em Estatística e Experimentação Agro-
pecuária, para a obtenção do título de “Doutor”.

Orientador

Prof. Dr. Júlio Sílvio de Sousa Bueno Filho

LAVRAS

MINAS GERAIS-BRASIL

2008

**Ficha Catalográfica Preparada pela Divisão de Processos Técnicos da
Biblioteca Central da UFLA**

Silva, José Waldemar da.

Algoritmos para modelos de limiar usando as distribuições acumuladas normal e t de Student / José Waldemar da Silva. -- Lavras : UFLA, 2008. 99 p. : il.

Tese (Doutorado) – Universidade Federal de Lavras, 2008.

Orientador: Júlio Sílvio de Sousa Bueno Filho.

Bibliografia.

1. Algoritmo Metropolis-Hastings. 2. MCMC. 3. Modelos lineares generalizados. 4. Modelos mistos. 5. Modelo de limiar. I. Universidade Federal de Lavras. II. Título.

CDD – 519.542

JOSÉ WALDEMAR DA SILVA

**ALGORITMOS PARA MODELOS DE LIMIAR USANDO AS DISTRIBUIÇÕES
ACUMULADAS NORMAL E t DE STUDENT**

de Federal de Lavras
o Programa de Pós-
perimentação Agro-
ítulo de “Doutor”.

APROVADA em 07 de março de 2008

Prof ^ª . Dr ^ª . Roseli Aparecida Leandro	ESALQ/USP
Prof ^ª . Dr ^ª . Joelma Pereira	UFLA
Prof ^ª . Dr ^ª . Thelma Sáfadi	UFLA
Prof. Dr. Daniel Furtado Ferreira	UFLA



Prof. Dr. Júlio Sílvio de Sousa Bueno Filho

UFLA

(Orientador)

LAVRAS

MINAS GERAIS-BRASIL

*Aos meus pais, **Gregório** (in memoriam) e **Maria Edith**,
por serem exemplos de determinação, sempre acreditarem
na educação de seus filhos e incentivá-los.*

*E aos meus irmãos, **Valdete**, **José Valdo**,
Valter, **Valdeir** e **Valdinei**,*

pelo carinho e

amizade,

Ofereço.

*À minha esposa **Elizângela**, pela amizade, compreensão e amor,*

Dedico.

AGRADECIMENTOS

A Deus, por tudo.

À Universidade Federal de Lavras (UFLA) e ao Programa de Pós-Graduação em Estatística e Experimentação Agropecuária por propiciar a realização deste trabalho.

Ao professor Júlio Sílvio de Sousa Bueno Filho pela orientação e ensinamentos.

Aos professores membros da banca examinadora, pelas críticas, sugestões e colaboração.

Aos professores do Departamento de Ciências Exatas (DEX), pelos ensinamentos fundamentais para a minha formação.

A todos os funcionários do DEX, pela atenção e eficiência com que nos atenderam.

Ao colega de doutorado, Janser Moura Pereira, pela amizade e companheirismo desde o início da caminhada no curso de graduação.

Aos colegas de curso pós-graduação, pelo incentivo, companheirismo e fraternidade, em especial aos amigos Imaculada, Claudiney, Verônica, Andréia, Denismar, Fabyano, Cirilo, Nádia, Rosi e Renato.

À Renata, pela amizade e por se dispor a me acolher sempre.

Aos amigos Célio e Célia, por me acolherem sempre prontamente em sua casa. Aos amigos Airton e Andréia por tudo e pelos momentos agradáveis que vivenciamos em Teresina-PI. Aos amigos, Isaías, Luciana e Larissa, Flávio e Carminha por me propiciarem momentos de descontração e alegria.

A todos os meus familiares, que contribuíram para a minha formação humana, acreditaram e colaboraram para o meu sucesso profissional e pessoal.

A todos aqueles que, direta ou indiretamente, contribuíram para a conclusão desta etapa, muito importante em minha vida, o meu sincero agradecimento.

SUMÁRIO

LISTA DE TABELAS	i
LISTA DE FIGURAS	iii
RESUMO	iv
ABSTRACT	vi
1 INTRODUÇÃO	1
2 REFERENCIAL TEÓRICO	4
2.1 Algoritmo de Sørensen et al. (1995) - (AC-normal)	8
2.2 Algoritmo de Cowles (1996) e Kizilkaya et al. (2003) - (MC-normal)	13
2.3 Algoritmo de Kizilkaya et al. (2003) - (MC-t)	15
2.4 Comparação de Modelos	22
3 MATERIAL E MÉTODOS	26
3.1 Material	26
3.2 Métodos	26
4 RESULTADOS E DISCUSSÃO	34
4.1 Resultados metodológicos	34
4.1.1 Algoritmo adaptado de AC (AC-t)	34
4.1.2 Algoritmo adaptado de NC - (NC-normal)	35
4.1.3 Algoritmo adaptado de NC - (NC-t)	40
4.2 Resultados do ajuste para os três algoritmos	44
4.2.1 Algoritmo AC (t e normal)	44
4.2.2 Algoritmo MC (t e normal)	50
4.2.3 Algoritmo NC (t e normal)	54
4.2.4 Análise dos contrastes de interesse	61
4.3 Comparações com a ANAVA e considerações gerais	62

5 CONCLUSÕES	66
REFERÊNCIAS BIBLIOGRÁFICAS	67
ANEXO	70

LISTA DE TABELAS

1	Escala hedônica de nove pontos.	4
2	Formas de análise encontradas na literatura para o problema de dados categorizados em escalas subjetivas.	21
3	Escala para interpretação do fator de Bayes.	24
4	Notas atribuídas por 36 provadores à cor da banana da terra desidratada utilizando três concentrações de sacarose.	27
5	Esquema de reclassificação dos dados em cinco categorias.	27
6	Valores de <i>Burn-in</i> (B) e <i>Thinning</i> (T) obtidos por meio do diagnóstico de Raftery & Lewis, para as médias de tratamentos (β), variância de blocos (σ_u^2), variância residuais (δ^2) e graus de liberdade (ν), nos algoritmos AC, MC e NC segundo a distribuição da variável latente.	47
7	Autocorrelação entre as estimativas para β_1 e para γ_3 no caso em que a função de ligação foi a distribuição normal acumulada e para β_2 e γ_2 quando a função de ligação foi a distribuição t de Student acumulada, no algoritmo AC.	49
8	Autocorrelação entre as estimativas de β_1 e de γ_4 no caso em que a função de ligação foi a distribuição normal acumulada e de β_1 e γ_3 quando foi usada a distribuição t de Student acumulada como função de ligação, no algoritmo MC.	53
9	Autocorrelação entre as estimativas de β_1 e de γ_2 no caso em que a função de ligação foi a distribuição normal acumulada e de β_1 e de γ_3 quando foi usada a distribuição t de Student acumulada como função de ligação, no algoritmo NC.	58

10	HPDs para funções dos parâmetros associadas à regressão linear $(C_1\beta)$, ao desvio de regressão $(C_2\beta)$ e à correlação intraclassa (ρ) .	62
11	Quadro-resumo da análise da variância para a variável cor sem transformação, com transformação logarítmica e com transformação Box-Cox.	64

LISTA DE FIGURAS

1	Funções de distribuição normal, t de Student e logística de acordo com o quantil.	16
2	Cadeias para médias de tratamentos no algoritmo AC e nas duas funções de ligação normal de t de Student, acumuladas.	45
3	Cadeias para os parâmetros de limiar no algoritmo AC e modelos normal e t de Student.	46
4	Cadeia e distribuição das estimativas de ν no algoritmo AC.	49
5	Cadeias para médias de tratamentos no algoritmo MC e nas duas funções de ligação normal de t acumuladas.	51
6	Cadeias para os parâmetros de limiar no algoritmo MC e modelos normal e t de Student.	52
7	Distribuição das estimativas de ν no algoritmo MC.	54
8	Cadeias para médias de tratamentos no algoritmo NC e nas duas funções de ligação normal de t de Student, acumuladas.	56
9	Cadeias para os parâmetros de limiar no algoritmo NC e modelos normal e t.	57
10	Cadeia e histograma para ν no algoritmo NC.	59
11	Cadeias para os contrastes associados à regressão linear e ao desvio de regressão no algoritmo AC e funções de ligação normal e t de Student, acumuladas.	63
12	Gráfico de resíduos a partir do modelo ajustado para a variável nota atribuída à cor da banana da terra desidratada, sem transformação (a), com transformação logarítmica (b) e com transformação de Box-Cox (c).	65

RESUMO

Silva, José Waldemar da. **Algoritmos para modelos de limiar usando as distribuições acumuladas Normal e t de Student.** Lavras: UFLA, 2008. 99 p. Tese (Doutorado em Estatística e Experimentação Agropecuária) - Universidade Federal de Lavras, Lavras, MG. *

Dados categorizados podem ser modelados por meio de uma variável latente (L) com escala contínua que relaciona a nota de uma categoria à pertinência em intervalo correspondente. Modela-se então a esperança de L por meio de uma função de ligação que é uma diferença entre valores de uma função acumulada nos limites de intervalo. Isto permite o estabelecimento de modelos lineares para a variável latente e propicia maior simplicidade computacional e facilidade de interpretação. O amostrador de Gibbs (GS) foi o primeiro algoritmo proposto (Albert & Chib, 1993; algoritmo AC) para a obtenção de amostras das distribuições conjuntas a *posteriori*, usando a distribuição t de Student, para o modelo fixo. Foi constatada, no entanto, forte autocorrelação nas cadeias dos parâmetros do modelo linear e nas dos parâmetros de limiar (limites dos intervalos para L). Visando minimizar este problema surgiram dois tipos de propostas de amostragem conjunta da variável latente e dos parâmetros de limiar usando algoritmos do tipo Metropolis-Hastings. A primeira usa como geradora de candidatos a distribuição normal truncada e está aqui implementada em um algoritmo modificado de Cowles (1996) (algoritmo MC). A segunda usa a distribuição Dirichlet e está no algoritmo modificado de Nandran & Chen (1996) (algoritmo NC). Neste trabalho as três metodologias para a modelagem da variável latente foram adaptadas para a análise de modelos mistos gerais. Os três algoritmos foram estudados quanto à rapidez de convergência em um experimento real de análise sensorial. Neste experimento, foi estudado o efeito de três concentrações de sacarose no processo de desidratação da banana da terra. Foram atribuídas notas para a cor do produto em uma escala hedônica de nove pontos. Quanto à convergência, a superioridade do algoritmo NC foi evidente para as duas funções de ligação utilizadas. O modelo com distribuição t de Student foi o melhor para os algoritmos AC e MC. Para o caso do algoritmo NC o resultado não discrimina os modelos, o que deve favorecer o modelo normal. Os resultados são semelhantes aos registrados por análises da variância usuais, cujas pressuposições são, no entanto, violadas. Os algoritmos estão implementados em R e permitem analisar quaisquer estruturas de covariância para os efeitos aleatórios.

Palavras-chave: Algoritmo Metropolis-Hastings; MCMC; modelos lineares generalizados; modelos mistos; modelos de limiar.

*Orientador: Júlio Sívio de Sousa Bueno Filho - UFLA.

ABSTRACT

Silva, José Waldemar da. **Algorithms for threshold models using Normal and Student-t cumulative distributions.** Lavras: UFLA, 2008. 99 p. Thesis (Doctorate in Statistics and Agricultural Experimentation) - Universidade Federal de Lavras, Minas Gerais, Brazil. *

Discrete ordinal data can be modeled by a latent variable (L) whose domain is a set of real intervals with bijecting relation between category and belonging to the respective interval. The expectation of L is then modeled using a link function that is a difference of cumulative distributions. This methodology has the advantage of establishing linear models for the latent variable with great computing simplification and interpretability. Gibbs Sampling (GS) was the first proposed algorithm to sample from the posterior in these situations (Albert & Chib, 1993; algorithm AC). This algorithm allows Student's t distribution, for fixed models. However, strong autocorrelations were found in Markov chains from both of the linear predictors and threshold parameters. Two kinds of proposals, both based on joint sampling of L and threshold parameters via Metropolis-Hastings type algorithms, are described to minimize this problem. The first uses a truncated normal distribution as a candidate generating function and is implemented here in an algorithm modified from Cowles (1996) (MC algorithm). The second uses a Dirichlet distribution and is implemented in the algorithm modified from Nandran & Chen (1996) (NC algorithm). In this work, the three methodologies were adapted to a general mixed model analysis. Convergence readiness in a real experiment was compared for the three algorithms. A sensorial analysis was used as an example. In this experiment, the effect of three concentrations of sacharosis in the dehydrating a banana variety were studied. Grades to the product color were given in a nine point hedonic scale. Regarding convergence, NC was the best algorithm for both normal and t distributions. Student's t was the more probable model using SAGK and MC algorithms, but for NC algorithm there was no evidence of difference among t and normal models (result that favors normal model). Resulting analysis were similar to usual ANOVA approximation, although violating its assumptions. Algorithms are implemented in R and allow to analyze a completely general covariance structure for random effects.

Key-words: MCMC; Metropolis-Hastings Algorithm; generalized linear models; mixed models; threshold models.

* Adviser: Júlio Sílvia de Sousa Bueno Filho - UFLA.

1 INTRODUÇÃO

Muitos atributos na experimentação, são avaliados por meios subjetivos. Juízes, técnicos, provadores ou consumidores são escolhidos ou designados para atribuir um conceito ao material experimental de acordo com suas preferências ou de acordo com algum regulamento.

Em geral, para a obtenção dos dados adota-se um conjunto de categorias dotadas de certa ordem nas quais os produtos possam ser classificados do pior ao melhor, de acordo com o julgamento de cada avaliador, diferentemente dos casos em que se tem uma variável contínua ou na forma de contagem.

Ainda é muito comum o uso desta metodologia sem a preocupação com a validade dos testes aplicados, apesar de os dados descritos não seguirem as pressuposições da análise de variância.

Outro ponto a ser destacado é a dificuldade ou impossibilidade encontrada, em muitos casos, para quantificar uma característica por meio de medição. Podem ser citados alguns exemplos típicos para esta situação como a seleção para algumas características morfológicas de touros, no melhoramento genético de gado de corte e, ou, leiteiro, tais como escore corporal e a classificação do animal quanto ao padrão racial, a avaliação do grau de dificuldade do parto de matrizes reprodutoras (vacas) e o grau de satisfação do consumidor com relação ao sabor de um produto alimentício a ser lançado no mercado. Vários outros exemplos poderiam ser mencionados, dada a grande quantidade de características de importância econômica existentes que são subjetivas, categóricas e ordinais.

A literatura recente sobre análise de dados categorizados subjetivamente, sugere o ajuste de modelos com parâmetros *threshold* ou de limiar, no qual, uma variável latente, com distribuição contínua, é especificada de forma que uma res-

posta é observada em uma dada categoria se o valor desta variável está entre os limites que definem tal categoria.

A variável latente facilita a implementação computacional, pois, pode ser modelada de forma linear. Esta variável pode ser interpretada como o estímulo que deve ser aplicado a um indivíduo, objeto ou produto, para que este possa ser classificado em uma das K categorias. A área da análise sensorial dos alimentos é um exemplo disto pois, muitas características são difíceis ou impossíveis de serem quantificadas por meio de medida objetiva. Nestes casos, cada provador ou juiz classifica um produto em alguma categoria, por meio do seu sabor, cor, textura, etc. Estas variáveis são dotadas de uma escala contínua, porém, a execução do experimento torna-se mais fácil quando são fragmentadas em categorias.

Wright (1934) foi o primeiro a propor esta metodologia, no estudo da herança do número de dedos de *Cavia porcellus* (porquinhos da Índia). Albert & Chib (1993) propuseram a utilização da amostragem Gibbs para obter aproximação das distribuições marginais dos parâmetros. Sørensen et al. (1995) utilizaram a mesma metodologia, porém na área da Genética Quantitativa Animal em que o Modelo Linear Generalizado (MLG) correspondente usava uma função de ligação normal acumulada.

A forte autocorrelação entre as estimativas, principalmente dos parâmetros de limiar, obtidos por meio da amostragem Gibbs, usando esta metodologia, é evidente e foi destacada por Sørensen et al. (1995) e Sørensen & Gianola (2004). Este fato obriga a obtenção de uma cadeia muito grande para que se tenha uma boa representação da distribuição marginal do parâmetro. Propostas de algoritmos para contornar ou amenizar este problema são dados por Cowles (1996) e Nandram & Chen (1996).

A escolha de um bom modelo e a conseqüente maior qualidade no processo

de inferência podem ser quantificados por alguns indicadores como a maior razão de verossimilhanças marginais dos modelos que se pretende comparar, a maior acurácia nas estimativas ou a melhor interpretabilidade de seus parâmetros.

O algoritmo para a análise de modelos mistos com função de ligação dada por uma distribuição normal acumulada foi desenvolvida por Sørensen et al. (1995) e é apresentada na seção 2.1. O mesmo algoritmo porém, para modelos fixos com função de ligação t de Student acumulada, é apresentado por Albert & Chib (1993). Contudo, outro algoritmo utilizando a amostragem conjunta dos parâmetros *threshold* e da variável latente foi proposto por Cowles (1996) e adaptado por Kizilkaya et al. (2003) para modelos mistos no melhoramento animal. Este algoritmo é apresentado nas seções 2.2, com a distribuição normal padrão como função de ligação, e 2.3, com t de Student acumulada. Ainda outra melhoria nestes algoritmos é a proposta de Nandran & Chen (1996) que visa reparametrizar os *threshold* e a variável latente.

Este trabalho teve como objetivo adaptar o algoritmo de Albert & Chib (1993) e de Nandran & Chen (1996) para os modelos mistos, além de usar a distribuição t de Student como função de ligação no segundo e contrastar a velocidade de convergência dos algoritmos propostos por Albert & Chib (1993) - AC, Cowles (1996) - MC e Nandram & Chen (1996) - NC e adaptados. A aplicação da metodologia será feita em um experimento de análise sensorial de alimentos.

Um objetivo adicional foi o de implementar estas metodologias em rotinas flexíveis no pacote estatístico R, que permitam análise de modelos mistos em situações frequentes e com generalidade suficiente para incluir estruturas de efeitos aleatórios correlacionados. Tais rotinas estão disponibilizadas em anexo.

TABELA 1: Escala hedônica de nove pontos.

Categorias	E. H. de 9 Pontos
Desgostei muitíssimo	1
Desgostei muito	2
Desgostei	3
Desgostei moderadamente	4
Nem gostei nem desgostei	5
Gostei moderadamente	6
Gostei	7
Gostei muito	8
Gostei muitíssimo	9

Fonte: Dutcosky (1996).

2 REFERENCIAL TEÓRICO

Alguns atributos na análise sensorial de alimentos são avaliados por meio da escala hedônica de nove pontos, conforme Tabela 1.

Porém, a existência de um grande número de categorias para a classificação pode dificultar a decisão do provador quanto a qual nota atribuir ao produto. Além disso, o acréscimo do número de parâmetros de limiar estimados pode dificultar a convergência do algoritmo e sob o aspecto da representatividade, Kizilkaya et al. (2003) sugerem o agrupamento de categorias com poucas observações.

O conhecimento prévio sobre um fenômeno e a informação fornecida por experimentos podem ser combinados por meio do Teorema de Bayes, usando a seguinte expressão:

$$p(\theta|Y) \propto p(\theta)p(Y|\theta),$$

em que θ é o vetor de parâmetros, Y o vetor de observações, $p(\theta|Y)$ a distribuição conjunta a *posteriori* e $p(\theta)$ a distribuição a *priori*.

As inferências são realizadas a partir da distribuição conjunta a *posteriori*

que em geral tem forma complexa e se torna difícil a avaliação das quantidades de interesse. O problema a ser resolvido é o cálculo da esperança a *posteriori* de alguma função $g(\theta)$,

$$E[g(\theta)|Y] = \int g(\theta)p(\theta|Y)d\theta, \quad (1)$$

A partir de uma amostra $\theta_1, \theta_2, \dots, \theta_n$ da distribuição a *posteriori* $p(\theta|Y)$ a integral em (1) é aproximada por meio da média,

$$\hat{E}[g(\theta)|Y] = \frac{1}{n} \sum_{i=1}^n g(\theta_i).$$

Geralmente este problema é transferido para o caso univariado e assim, ele seria solucionado calculando-se a distribuição marginal para cada parâmetro. No entanto, a abordagem univariada não contorna a dificuldade de integração para a obtenção da esperança a *posteriori*. Métodos numéricos como a amostragem Gibbs podem ser usados para aproximar a integral em (1) quando as distribuições condicionais completas a *posteriori* possuem formas algébricas fechadas (das quais se pode tomar amostras). Caso isso não ocorra, deve-se lançar mão de outras formas de amostragem, em geral baseadas no algoritmo Metropolis-Hastings (Metropolis et al., 1953 e Hastings, 1970).

Os modelos lineares generalizados (GLM) podem ser usados para estabelecer uma relação entre uma variável resposta categorizada ordinal e variáveis preditoras, sendo que, a função ligadora é alguma função de distribuição. Supondo que existam n variáveis, classificadas em K categorias ordenadas Y_1, Y_2, \dots, Y_n , em que Y_i é tal que $p_{ik} = P(Y_i = k)$ para $k = 1, 2, \dots, K$ e $\sum_{k=1}^K p_{ik} = 1, i = 1, \dots, n$ e assumindo que as probabilidades acumuladas possam ser modelada por

um GLM, têm-se:

$$P(Y_i \leq k) = G(\gamma_k - w_i' \theta), k = 1, 2, \dots, K \quad (2)$$

em que w_i' é o vetor linha de incidência, ligando θ à i -ésima observação, G é uma função de distribuição de uma variável contínua que pode assumir várias formas, θ é o vetor de efeitos fixos e aleatórios e γ é o vetor de parâmetros *threshold* que divide a reta real em K intervalos disjuntos, da seguinte forma:

$$(\gamma_0, \gamma_1); [\gamma_1, \gamma_2); \dots; [\gamma_{K-1}, \gamma_K);$$

em que $\gamma_0 = -\infty$ e $\gamma_K = +\infty$

Em geral adota-se uma origem para a variável latente fixando um dos parâmetros de limiar ou *threshold* e em geral, escolhe-se $\gamma_1^* = 0$. A motivação para $\gamma_1^* = 0$ é dada pela transformação $\gamma_k^* = \gamma_k - \gamma_1$. Assim, cada γ_k^* será a distância entre γ_k e γ_1 . A variável latente transformada será obtida por meio da expressão $L_i^* = L_i - \gamma_1, i = 1, \dots, n$ e assim,

$$y_i = k \text{ se } \gamma_{k-1} - \gamma_1 \leq L_i - \gamma_1 \leq \gamma_k - \gamma_1$$

Os novos parâmetros (θ_i^*), de efeitos fixos e aleatórios, serão $\theta_i^* = \theta_i - \gamma_1, i = 0, 1, \dots, m$.

Na escala da variável latente, $L^* = W\theta^* + \epsilon$ em que, W é a matriz de incidência, θ^* o vetor de efeitos e ϵ o vetor de resíduos. Assumindo distribuição normal para a variável latente (L^*), então

$$(L^* | \theta^*, \sigma_\epsilon^2) \sim N(W\theta^*, I\sigma_\epsilon^2).$$

Sem perda de generalidade a variância residual pode ser constante, adotando-se

por exemplo, $\sigma_\epsilon^2 = 1$ (Sørensen & Gianola, 2004).

A distribuição conjunta a *posteriori*, de todos os parâmetros e da variável latente resulta em:

$$\begin{aligned}
 p(\theta^*, L^*, \gamma^* | Y) &\propto p(Y | \theta^*, L^*, \gamma^*) p(\theta^*, \gamma^*) = p(Y | L^*, \gamma^*) p(\theta^*, L^*, \gamma^*) \\
 &= p(Y | L^*, \gamma^*) p(\theta^*, L^*, \gamma^*) = p(Y | L^*, \gamma^*) p(L | \theta^*) p(\theta^*, \gamma^*) \\
 &= p(Y | L^*, \gamma^*) \left[\prod_{i=1}^n p(L_i^* | \theta^*) \right] p(\theta^*, \gamma^*) \quad (3)
 \end{aligned}$$

A distribuição $p(Y | L^*, \gamma^*)$ em (3) é degenerada pois, a probabilidade de uma determinada observação pertencer a uma determinada categoria, dado o valor da variável latente e os limiares, é completamente especificada. Conforme notação de Albert & Chib (1993), $p(Y | L^*, \gamma^*)$ pode ser escrita como

$$p(Y | L^*, \gamma^*) = \prod_{i=1}^n \left[\sum_{k=1}^K I_{[\gamma_{k-1}^*, \gamma_k^*)}(L_i^*) I(Y_i = k) \right].$$

Albert & Chib (1993) apresentam um algoritmo para a modelagem de dados categorizados, no qual a função de ligação (G) em (2) corresponde à distribuição t de Student acumulada. O mesmo algoritmo é implementado por Sørensen et al. (1995) em que a função de ligação em (2) é a distribuição normal padrão acumulada considerando o modelo misto (efeitos fixos e aleatórios no modelo para a variável latente). Nesta modelagem, um parâmetro de limiar ou *threshold* (γ_k^*) tem distribuição condicional completa uniforme com domínio no intervalo $[\max(L^* | Y = k); \min(L^* | Y = k + 1)]$ em que L^* é o vetor de variáveis latentes.

O algoritmo desenvolvido por Sørensen et al. (1995) é descrito a seguir.

2.1 Algoritmo de Sørensen et al. (1995) - (AC-normal)

Supondo que a função G em (2) é dada por uma distribuição normal padrão acumulada ($\Phi(\cdot)$), então o modelo amostral é

$$p(Y_i = k | \theta^*, \gamma^*) = \Phi(\gamma_k^* - w_i' \theta^*) - \Phi(\gamma_{k-1}^* - w_i' \theta^*). \quad (4)$$

e usando uma variável latente L^* podemos reescrever (2) dados os parâmetros de limiar γ^* , θ^* e w_i como:

$$Y_i = k \text{ se } \gamma_{k-1}^* \leq L_i \leq \gamma_k^*$$

$$L_i^* \sim N(w_i' \theta^*, 1) \quad (5)$$

com $i = 1, 2, \dots, n$ e $k = 1, 2, \dots, K$.

Para a variável latente L^* , tem-se o modelo linear $L^* = W\theta^* + \epsilon$, em que $W = [X|Z]$, com X e Z correspondendo às matrizes de delineamento para os efeitos fixos e aleatórios, respectivamente; ϵ é o vetor de erros aleatórios e $\theta^* = (\beta', u')$ é o vetor de efeitos fixos (β) e aleatórios (u). Quando não se tem efeitos aleatórios o modelo terá sua estrutura simplificada, com $W = X$ e $\theta^* = \beta'$.

Assumindo que os elementos de Y são condicionalmente independentes dado θ^* , a densidade conjunta a *posteriori* de todos os parâmetros, incluindo a variável latente L^* , é dada como segue:

$$\begin{aligned} p(\theta^*, \gamma^*, L^* | Y) &\propto p(Y, L^*, \theta^*, \gamma^*) = p(Y, L^* | \theta^*, \gamma^*) p(\theta^*, \gamma^*) \\ &= p(Y | L^*, \theta^*, \gamma^*) p(L^* | \theta^*, \gamma^*) p(\theta^*, \gamma^*) \\ &\propto \left[\prod_{i=1}^n \exp\left(-\frac{1}{2}(L_i^* - w_i' \theta^*)^2\right) \right] I_{[\gamma_{k-1}^*, \gamma_k^*)}(L_i^*) \phi(\theta^*; \mathbf{0}, V) \end{aligned} \quad (6)$$

A informação a priori sobre (θ^*, γ) pode ser dada por:

$$p(\theta^*, \gamma) \propto p(\theta^*) \propto \exp \left\{ -\frac{\tau}{2} \theta^{*'} \theta^* \right\}$$

a qual será vaga se o parâmetro de precisão τ assumir um valor pequeno, por exemplo, $\tau = 0,001$ ou em (6) fazendo $V = 1000I_m$, em que I_m é uma matriz identidade $m \times m$.

Quando se tem um modelo linear misto, com $\theta' = (\beta', u')$, u é especificado por

$$u | \sigma_u^2 \sim N(\emptyset, A_q \sigma_u^2)$$

em que A é a matriz de correlação entre os efeitos aleatórios. Em modelos de Genética Quantitativa, por exemplo, A é um múltiplo da matriz de parentesco (Kempthorne, 1966).

Assim, a matriz V pode ser dada por

$$V = \begin{pmatrix} 1000I_p & \emptyset \\ \emptyset & \sigma_u^2 A_q \end{pmatrix}$$

em que σ_u^2 é a variância dos efeitos aleatórios, p e q correspondem, respectivamente, à quantidade de efeitos fixos e à quantidade de efeitos aleatórios, I_p é uma matriz identidade $p \times p$ e A_q é a matriz de correlação entre os elementos de u , $q \times q$.

No modelo misto,

$$p(\theta^*, \gamma) \propto p(\theta^*) \propto \left(\frac{1}{\sigma_u^2} \right)^{\binom{q}{2}} \exp \left\{ -\frac{1}{2} \theta^{*'} V^{-1} \theta^* \right\}.$$

A *priori* para σ_u^2 pode ser dada por uma Gama Inversa,

$$p(\sigma_u^2) = (\sigma_u^2)^{-(a+1)} \exp\left\{\frac{-b}{\sigma_u^2}\right\}, \quad (7)$$

em que a e b são os parâmetros desta distribuição *a priori*, a qual tem desejáveis propriedades de conjugação com os modelos normais.

E assim, a distribuição conjunta em (6) é reescrita como segue:

$$\begin{aligned} p(\theta^*, \gamma^*, L^*, \sigma_u^2 | Y) &\propto p(Y, L^*, \theta^*, \gamma^*) = p(Y, L^* | \theta^*, \gamma^*) p(\theta^*, \gamma^*) \\ &= p(Y | L^*, \theta^*, \gamma^*) p(L^* | \theta^*, \gamma^*) p(\theta^*, \gamma^*) \\ &\propto \left[\prod_{i=1}^n \exp\left(-\frac{1}{2}(L_i^* - w_i \theta^*)^2\right) \right] I_{[\gamma_{k-1}^*, \gamma_k^*)}(L_i^*) \\ &\times \phi(\theta^*; \mathbf{0}, V) p(\sigma_u^2) \end{aligned} \quad (8)$$

As inferências sobre cada parâmetro podem ser feitas com base em amostras da distribuição conjunta *a posteriori* (Gelfand & Smith, 1990), calculadas a partir de (6), se o interesse for no modelo fixo ou a partir de (8) se o interesse for no modelo misto, usando os métodos Monte Carlo via Cadeias de Markov (MCMC).

A metodologia de amostragem das distribuições condicionais foi originalmente desenvolvida por Sørensen et al. (1995) a partir de (8) e está apresentada também, de forma bem didática, em Sørensen & Gianola (2004). Nos parágrafos que se seguem estamos reproduzindo a maior parte desta apresentação.

É preciso notar que a distribuição $p(Y | L^*, \gamma^*)$ na expressão em (8) é degenerada, pois o conhecimento de L^* leva ao conhecimento exato de Y (Albert &

Chib, 1993) e sua expressão é dada por:

$$p(Y|L^*, \gamma^*) = \prod_{i=1}^n \left\{ \sum_{k=1}^K I_{[\gamma_{k-1}^*, \gamma_k^*]}(L_i^*) I(Y_i = k) \right\} \quad (9)$$

A partir da expressão anterior pode-se observar que a distribuição condicional completa de γ_k^* dado γ_{-k}^*, θ^* e L^* , em que γ_{-k}^* representa todos os parâmetros de limiar exceto γ_k , é dada por:

$$\begin{aligned} P(\gamma_k^* | \gamma_{-k}^*, L^*, Y) &\propto \\ &\propto \prod_{i=1}^n \left[I(Y_i = k) I_{[\gamma_{k-1}^*, \gamma_k^*]}(L_i^*) + I(y_i = k + 1) I_{[\gamma_k^*, \gamma_{k+1}^*]}(L_i^*) \right] \end{aligned} \quad (10)$$

a qual pode ser vista como uma distribuição uniforme, conforme expressão a seguir

$$P(\gamma_k^* | \gamma_{-k}^*, L^*, Y) = \frac{1}{a - b}, \quad a > b \quad (11)$$

em que $a = \min((L^* | Y = k + 1), \gamma_{k+1}^*)$ e $b = \max((L^* | Y = k), \gamma_{k-1}^*)$.

Com relação à variável latente, observa-se que L_i^* tem distribuição condicional completa a *posteriori* dada por:

$$P(L_i^* | \theta^*, Y_i = k, \gamma_{k-1}^*, \gamma_k^*) = \frac{\phi_{L_i^*}(w_i' \theta^*, 1)}{\Phi(\gamma_k^* - w_i' \theta^*) - \Phi(\gamma_{k-1}^* - w_i' \theta^*)}, \quad (12)$$

com $\gamma_{k-1}^* < L_i^* \leq \gamma_k^*$, $\Phi(\cdot)$ representando a distribuição normal padrão acumulada e $\phi_{L_i^*}(w_i' \theta^*, 1)$ a densidade normal com média $w_i' \theta^*$ e variância unitária.

O vetor de parâmetros θ^* tem distribuição condicional completa a *posteri-*

ori dada por uma normal multivariada, ou seja,

$$\theta^* | Y, L^* \sim N \left(B^{-1} W' L^*, B^{-1} \right) \quad (13)$$

em que $B = V^{-1} + W'W$.

Para a variância dos efeitos aleatórios (σ_u^2), a distribuição condicional completa será

$$p \left(\sigma_u^2 | \theta^*, L^*, Y \right) = \left(\sigma_u^2 \right)^{-\left(\frac{q}{2} + a + 1\right)} \exp \left\{ \frac{-1}{2\sigma_u^2} \left(u' u + 2b \right) \right\}, \quad (14)$$

que é o núcleo de uma distribuição Gamma Inversa com parâmetros como a seguir:

$$\sigma_u^2 | \theta^*, L^*, Y \sim GI \left(\frac{q + 2a}{2}, \frac{u' u + 2b}{2} \right) \quad (15)$$

A amostragem Gibbs, segundo Sørensen et al. (1995), pode ser implementada a partir das distribuições condicionais completas dadas em (13), (15), (12) e (11), não necessariamente nesta ordem.

Um problema apontado por alguns autores como Sørensen & Gianola (2004), Cowles (1996), Nandran & Chen (1996) e Kizilkaya et al. (2003) é que nos algoritmos apresentados por Albert & Chib (1993) e por Sørensen et al. (1995) existe uma forte autocorrelação entre os valores obtidos por meio da amostragem Gibbs, principalmente para os parâmetros de limiar. Segundo Nandran & Chen, (1996) dificuldade ou lentidão na convergência para esses parâmetros acarreta, por consequência, dificuldade de convergência para θ .

Algumas propostas de algoritmos para contornar ou amenizar este problema são dados por Cowles (1996), Kizilkaya et al. (2003) e Nandran & Chen (1996).

Para contornar ou amenizar problemas de convergência Cowles (1996) apresenta uma análise em que amostras da distribuição conjunta a *posteriori* são obtidas por meio da amostragem Gibbs com um “passo” Metropolis-Hastings para obter amostras dos parâmetros *threshold* e da variável latente a partir da forma fatorada $p(\gamma^*, L^* | \theta^*, Y) = p(L^* | \gamma^*, \theta^*, Y)p(\gamma^* | \theta^*, Y)$. Nesta análise a função de ligação utilizada em (2) foi a distribuição normal padrão acumulada e um modelo fixo para a variável latente. De acordo com o método da composição, apresentado por Devroye (1986, citado por Gelfand & Smith, 1990), Chib (2001) e Sørensen & Gianola (2004), a distribuição de $(\gamma^*, L^* | \theta^*, Y)$ pode ser escrita como o produto das distribuições condicionais $(\gamma^* | \theta^*, Y)$ e $(L^* | \gamma^*, \theta^*, Y)$, conforme teorema de Bayes em que $(L^* | \gamma^*, \theta^*, Y)$ é obtida a partir da *posteriori* conjunta e $(\gamma^* | \theta^*, Y)$ é obtida da integração de $(\gamma^*, L^* | \theta^*, Y)$ em relação à L^* . Conforme Liu et al. (1994) agrupando os elementos a serem amostrados, em geral, tem-se aumento na eficiência do amostrador de Gibbs. A candidata utilizada para gerar valores de γ_k^* nesta metodologia foi a distribuição normal, truncada por outros dois parâmetros *threshold* adjacentes, com média dada pelo *threshold* amostrado na iteração anterior e uma variância fixa.

Seguindo a metodologia apresentada por Cowles (1996), Kizilkaya et al. (2003) apresentam um modelo misto para a variável latente associada à variável *score* para facilidade de parto, utilizando como funções de ligação as distribuições normal e t de Student acumuladas.

2.2 Algoritmo de Cowles (1996) e Kizilkaya et al. (2003) - (MC-normal)

Visando melhorar o processo de amostragem, é apresentada a seguir, a proposta de Kizilkaya et al. (2003), adaptada de Cowles (1996). Segundo esta metodologia, a atualização dos parâmetros de limiar não é feita de forma isolada,

como proposto por Albert & Chib (1993), mas de forma a aceitar todo o vetor formado por estes parâmetros.

A distribuição para $(\gamma^*, L^*|\theta^*, Y)$ pode ser fatorada como o produto das distribuições condicionais $(\gamma^*|\theta^*, Y)$ e $(L^*|\gamma^*, \theta^*, Y)$. A distribuição condicional para $(\gamma^*|\theta^*, Y)$ é obtida a partir da integral de $(\gamma^*, L^*|\theta^*, Y)$ em relação a L^* :

$$\begin{aligned}
 p(\gamma^*|\theta^*, Y) &\propto \prod_{Y_i=2} \left[\Phi(\gamma_2^* - w_i'\theta^*) - \Phi(-w_i'\theta^*) \right] \\
 &\times \prod_{Y_i=3} \left[\Phi(\gamma_3^* - w_i'\theta^*) - \Phi(\gamma_2^* - w_i'\theta^*) \right] \dots \\
 &\dots \times \prod_{Y_i=K-1} \left[\Phi(\gamma_{K-1}^* - w_i'\theta^*) - \Phi(\gamma_{K-2}^* - w_i'\theta^*) \right] \\
 &\times \prod_{Y_i=K} \left[1 - \Phi(\gamma_{K-1}^* - w_i'\theta^*) \right] \tag{16}
 \end{aligned}$$

e a distribuição de $(L^*|\gamma^*, \theta^*, Y)$ resulta em:

$$L^*|\gamma^*, \theta^*, Y_i = k \sim N(w_i'\theta^*, 1), \tag{17}$$

com $\gamma_{k-1}^* < L_i^* \leq \gamma_k^*$.

Observe que para a amostragem de γ^* é necessário o uso do algoritmo Metropolis-Hastings pois estes parâmetros não têm uma distribuição distribuição da qual se possa amostrar diretamente.

Cowles (1996) propôs usar a distribuição normal, para a geração de candidatos para um determinado *threshold*, truncada por outros dois adjacentes. Isto é,

$$\gamma_k^*|\gamma_{k-1,j}^*, \gamma_{k+1}^{*(j-1)} \sim N\left(\gamma_k^{*(j-1)}, \sigma_\gamma^2\right). \tag{18}$$

em que j representa a j -ésima iteração e $\gamma_{k-1,j}^* \leq \gamma_k^* < \gamma_{k+1}^{*(j-1)}$. A probabilidade

de aceitação do novo vetor de parâmetros *threshold* é o mínimo entre 1 e R, em que

$$R = \prod_{k=2}^{K-1} \frac{\Phi \left\{ \left(\gamma_{k+1}^{*(j-1)} - \gamma_k^{*(j-1)} \right) / \sigma_\gamma \right\} - \Phi \left\{ \left(\gamma_{k-1,j}^* - \gamma_k^{*(j-1)} \right) / \sigma_\gamma \right\}}{\Phi \left\{ \left(\gamma_{k+1,j}^* - \gamma_{k,j}^* \right) / \sigma_\gamma \right\} - \Phi \left\{ \left(\gamma_{k-1}^{*(j-1)} - \gamma_{k,j}^* \right) / \sigma_\gamma \right\}} \times \prod_{i=1}^n \frac{\Phi \left(\gamma_{Y_i,j}^* - w_i' \theta^* \right) - \Phi \left(\gamma_{Y_{i-1},j}^* - w_i' \theta^* \right)}{\Phi \left(\gamma_{Y_i}^{*(j-1)} - w_i' \theta^* \right) - \Phi \left(\gamma_{Y_{i-1}}^{*(j-1)} - w_i' \theta^* \right)} \quad (19)$$

O amostrador de Gibbs pode ser implementado a partir de (13), (15), (18) e (17).

Segundo Albert & Chib (1993), inferências podem ser sensível à função de ligação adotada para G em (2). Alternativamente à distribuição normal, Albert & Chib (1993), Kizilkaya et al. (2003) atribuíram distribuição t de Student para a variável latente. A distribuição logística investigada por Albert & Chib, (1993) na análise de dados dicotomizados é outra alternativa (Figura 1).

2.3 Algoritmo de Kizilkaya et al. (2003) - (MC-t)

Em geral a distribuição t de Student é considerada mais plástica para dados binários, mas no caso dos dados categorizados, é provável que tal distribuição possa assumir formas mais próximas tanto da distribuição normal quanto da logística, com diferentes gradações dadas pelo número de graus de liberdade. A distribuição t de Student aproxima-se da distribuição normal quando o número de graus de liberdade ν é grande ($\nu \rightarrow \infty$) e da distribuição logística quando ($\nu \rightarrow 0$). Na Figura (1) estão ilustradas as distribuições normal padrão, t de Student com 3 graus de liberdade e logística.

Para a utilização da distribuição t de Student acumulada em (2) é interes-

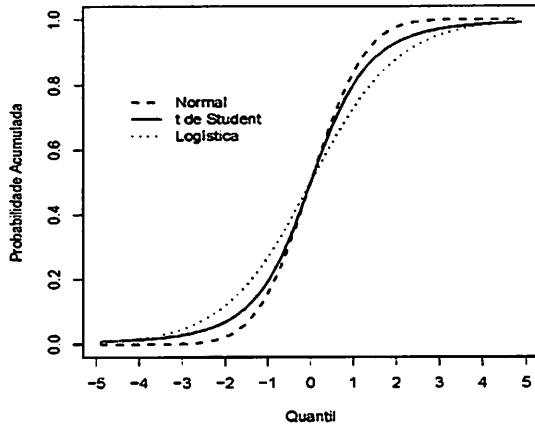


FIGURA 1: Funções de distribuição normal, t de Student e logística de acordo com o quantil.

sante escrevê-la em dois estágios (Paulino et al., 2003, Sørensen & Gianola 2004), isto é, como a mistura de normais (21) com a gamma invertida (22) para a variância. Tendo em vista que a maioria das distribuições utilizadas na modelagem são da família exponencial, este procedimento propiciará facilidades algébricas e melhor tratabilidade analítica para as condicionais completas a posteriori.

Utilizando densidade acumulada de uma distribuição t de Student com ν graus de liberdade a probabilidade de Y pode ser modelada por meio da expressão (20).

$$P(Y_i = k|\theta^*, \nu, \gamma^*) = F_\nu(\gamma_k^* - w_i'\theta^*) - F_\nu(\gamma_{k-1}^* - w_i'\theta^*), \quad (20)$$

$$k = 1, 2, \dots, K.$$

Amstras da distribuição t de Student para a variável latente podem ser

obtidas por um processo hierárquico de amostragem de distribuição normal com variância modificada. Desta forma, amostra-se inicialmente os parâmetros modificadores λ a partir da distribuição em (22) e em seguida, utilizando estes valores, amostra-se L de (21).

Os dois estágios para a especificação da distribuição t de Student são indicados nas expressões (21) e (22).

$$L_i^* | \theta^*, \lambda_i \sim N \left(w_i' \theta^*, \frac{1}{\lambda_i} \right), i = 1, 2, \dots, n \quad (21)$$

$$\lambda_i | \nu \sim \text{Gamma} \left(\frac{\nu}{2}, \frac{\nu}{2} \right). \quad (22)$$

Com esta especificação, o modelo amostral (20) pode ser reescrito da seguinte maneira:

$$P(Y_i = k | \theta^{**}, \lambda_i, \nu, \gamma^{**}) = \Phi \left(\frac{\gamma_k^{**} - w_i' \theta}{\frac{1}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{k-1}^{**} - x_i' \theta}{\frac{1}{\sqrt{\lambda_i}}} \right), \quad (23)$$

$k = 1, 2, \dots, K$

Admitindo *priori* vaga para θ^* , conforme especificado em 2.1, a posteriori conjunta de todos os parâmetros e a variável auxiliar ou latente é:

$$\begin{aligned} p(\theta^*, \gamma_k^*, L^*, \lambda, \sigma_u^2, \nu | Y) &\propto \left[\prod_{i=1}^n \phi \left(L_i^*, w_i \theta^*, \frac{1}{\lambda_i} \right) I_{[\gamma_{k-1}^*, \gamma_k^*]}(L_i^*) \right] \\ &\times \left[\prod_{i=1}^n \lambda_i^{\left(\frac{\nu}{2}\right)-1} \exp \left(-\frac{\lambda_i}{2} \nu \right) \right] \\ &\times \phi(\theta^*, 0, V) p(\sigma_u^2) p(\nu) \end{aligned} \quad (24)$$

em que $\lambda = \{\lambda_i\}_{i=1}^n$.

A matriz V , em (24) é dada na seção 2.1, a *priori* para σ_u^2 , é como em

(7) e a priori para ν é $p(\nu) = 1/(1 + \nu)^2$, isto representa pequena probabilidade de ocorrência de valores altos para os graus de liberdade, já que, para ν grande a distribuição t de Student aproxima-se da distribuição normal.

Da mesma forma com que foi descrito para a distribuição normal, a amostragem de γ^* e L^* é feita de forma conjunta. Quando a função de ligação é a distribuição t de Student acumulada, a distribuição de $(\gamma^*|\theta^*, Y)$ resulta em:

$$\begin{aligned}
 p(\gamma^*|\theta^*, Y) &\propto \prod_{Y_i=2} \left[\Phi \left(\frac{\gamma_2^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{-w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) \right] \\
 &\times \prod_{Y_i=3} \left[\Phi \left(\frac{\gamma_3^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_2^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) \right] \dots \\
 &\dots \times \prod_{Y_i=K-1} \left[\Phi \left(\frac{\gamma_{K-1}^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{K-2}^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) \right] \\
 &\times \prod_{Y_i=K} \left[1 - \Phi \left(\frac{\gamma_{K-1}^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) \right]
 \end{aligned}$$

e L^* tem distribuição normal com a variância modificada por λ como segue:

$$L^*|\gamma^*, \theta^*, Y_i = k \sim N \left(W\theta^*, \frac{1}{\sqrt{\lambda_i}} \right). \quad (25)$$

A distribuição candidata para gerar os novos parâmetros de limiar também pode ser dada por uma normal como em (18) e a probabilidade de aceitação será o mínimo entre 1 e R, em que R é dado por:

$$\begin{aligned}
 R &= \prod_{k=2}^{K-1} \frac{\Phi \left\{ \left(\gamma_{k+1}^{*(j-1)} - \gamma_k^{*(j-1)} \right) / \sigma_\gamma \right\} - \Phi \left\{ \left(\gamma_{k-1}^* - \gamma_k^{*(j-1)} \right) / \sigma_\gamma \right\}}{\Phi \left\{ \left(\gamma_{k+1,j}^* - \gamma_{k,j}^* \right) / \sigma_\gamma \right\} - \Phi \left\{ \left(\gamma_{k-1}^{*(j-1)} - \gamma_{k,j}^* \right) / \sigma_\gamma \right\}} \\
 &\times \prod_{i=1}^n \frac{\Phi \left(\frac{\gamma_{Y_i,j}^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{Y_i-1,j}^* - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right)}{\Phi \left(\frac{\gamma_{Y_i}^{*(j-1)} - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{Y_i-1}^{*(j-1)} - w_i' \theta^*}{\frac{1}{\sqrt{\lambda_i}}} \right)}. \quad (26)
 \end{aligned}$$

A seguir são apresentadas as distribuições condicionais completas a partir de (24) para cada parâmetro.

Para o vetor θ a distribuição condicional resulta em:

$$\theta^* | Y, L^*, \lambda, \gamma^* \sim N \left(M^{-1} W' R^{-1} L^*, M^{-1} \right) \quad (27)$$

em que $M = W' R^{-1} W + V^{-1}$ e R é uma matriz diagonal com seus elementos dados por λ_i^{-1} .

A distribuição condicional completa para λ_i é dada conforme expressão a seguir:

$$p(\lambda_i | \lambda_{-i}, L^*, \theta^*, \nu) \propto \lambda_i^{\left(\frac{\nu+1}{2}\right)-1} \exp \left(-\frac{\lambda_i}{2} \left((L_i^* - w_i' \theta)^2 + \nu \right) \right) \quad (28)$$

em que λ_{-i} denota todos os elementos de λ , exceto λ_i . Verifica-se a partir de (28) que cada λ_i tem distribuição condicional proporcional a uma gamma com parâmetros dados abaixo.

$$\lambda_i | \lambda_{-i}, L^*, \theta^*, \nu \sim G \left(\frac{\nu+1}{2}, \frac{1}{2} \left((L_i^* - w_i' \theta)^2 + \nu \right) \right). \quad (29)$$

A distribuição condicional para L_i resultará em uma normal conforme a expressão:

$$L_i^* | \lambda_i, \theta^*, \nu \sim N \left(w_i' \theta^*, \frac{1}{\lambda_i} \right), \gamma_{k-1}^* < L_i^* < \gamma_k^*, k = 1, 2, 3, \dots, K. \quad (30)$$

A distribuição condicional completa para ν não tem forma fechada e assim, torna-se necessário o uso do algoritmo Metropolis-Hastings para a amostragem

desta distribuição

$$p(\nu|\theta^*, L^*, \gamma^*, \lambda) \propto \left(\frac{(\frac{\nu}{2})^{(\nu/2)}}{\Gamma(\nu/2)} \right)^n \left(\prod_{i=1}^n \lambda_i^{\frac{\nu}{2}-1} \exp\left(-\frac{\nu}{2}\lambda_i\right) \right) \frac{1}{(1+\nu)^2} \quad (31)$$

As distribuições condicionais para os parâmetros σ_u^2 e γ_k são, respectivamente, dadas pelas expressões (15) e (25).

Neste caso, a amostragem Gibbs é realizada por meio das condicionais completas dadas em (27), (15), (18), (29), (31) e (25), não necessariamente nesta ordem.

Dada a dificuldade de se obter um valor adequado para a variância da distribuição geradora de candidatos para (γ_k^*) , na metodologia proposta por Cowles (1996) e com o intuito de melhorar o processo de convergência, Nandran & Chen (1996) apresentam a reparametrização dada por:

$$\delta = 1/\gamma_{K-1}^*, \gamma_k^{**} = \delta\gamma_k^*, k = 0, 1, 2, \dots, K, \theta^{**} = \delta\theta^* \text{ e } L^{**} = \delta L^*. \quad (32)$$

A amostragem de γ^{**} e L^{**} é como em Cowles (1996). Com esta reparametrização, os parâmetros de limiar a serem estimados (γ_k^*) ficam limitados entre 0 e 1. Além disso, utilizando o teorema do valor médio, verifica-se que a distribuição de $(\gamma^*|\theta^*, Y)$ é dada de forma proporcional ao produto de uma distribuição normal e uma distribuição Dirichlet. A limitação dos parâmetros de limiar entre 0 e 1, motiva à utilização desta última como geradora de candidatos para estes parâmetros, ainda que indiretamente. Outro atrativo desta forma de gerar candidatos é que com esta distribuição o vetor dos valores propostos é gerado de uma só vez e não depende dos demais parâmetros. Nesta abordagem, a função de ligação considerada para (2) é a distribuição normal acumulada e um modelo fixo é adotado para L^{**} . Um dos nossos propósitos é adaptar a metodologia de Nandran &

TABELA 2: Formas de análise encontradas na literatura para o problema de dados categorizados em escalas subjetivas.

Função geradora de candidatos para γ	Distribuição da variável latente	Forma de análise	
		Modelos fixos	Modelos com efeitos aleatórios
Condicionais completas	Normal	Sörensem et al. (1995)	Sörensem et al. (1995)
Normal	t de student	Albert & Chib (1993)	
	Normal	Cowles (1996)	Kizilkaya et al.(2003)
	t de Student		Kizilkaya et al.(2003)
Dirichlet	Normal	Nandran & Chen (1996)	
	t de Student		

Chen (1996) para modelos mistos, com distribuições normal e t de Student para a variável latente.

Na Tabela 2 estão indicados os algoritmos, as distribuições adotadas para a variável latente, o tipo de modelo (fixo ou aleatório), bem como, a autoria encontrados na literatura. As células em branco na coluna, que se refere à forma de análise por meio de modelos com efeitos aleatórios, compreendem às metodologias que serão desenvolvidas neste trabalho. Nesta Tabela está indicado também que a amostragem Gibbs é utilizada nos algoritmos desenvolvidos por Sörensen et al. (1995) e Albert & Chib (1993), e nos outros dois a correspondente distribuição geradora de candidatos, pois, a distribuição conjunta da variável latente e dos parâmetros *threshold* é escrita de forma fatorada e faz-se necessário o uso do algoritmo Metropolis-Hastings.

As formas de análises apresentadas na Tabela 2 podem ser classificadas em três grupos quanto à amostragem. O primeiro utiliza a amostragem Gibbs, o segundo a amostragem por meio do algoritmo Metropolis-Hastings (MH) e distribuição normal como geradora de candidatos e o último, a amostragem utilizando o algoritmo MH e distribuição Dirichlet como geradora de candidatos. Estas três formas de amostragem foram originalmente desenvolvidas, respectivamente, por

Albert & Chib (1993), Cowles (1996) e Nandram & Chen (1996), como já mencionado. As outras autorias indicadas na Tabela 2 são adaptações quanto à distribuição adotada para a variável latente e modelos com efeitos aleatórios.

2.4 Comparação de Modelos

A comparação entre modelos pode ser realizada empregando o fator de Bayes (Gelman et al., 2003), o qual de forma geral é dado por:

$$B_{ij} = \frac{p(y|M_i)}{p(y|M_j)} = \frac{\int p(y|\theta_i, M_i)p_i(\theta_i|M_i)d\theta_i}{\int p(y|\theta_j, M_j)p_j(\theta_j|M_j)d\theta_j} \quad (33)$$

em que θ_i e θ_j são os vetores de parâmetros dos modelos M_i e M_j , respectivamente, com M_i e M_j dois modelos concorrentes. Se o fator (B_{ij}) for maior do que 1 há evidências de que o modelo i seja melhor do que o modelo j .

Porém, nem sempre é possível obter as integrais indicadas em (33), alternativamente o método da amostragem por importância pode ser considerado para obter uma aproximação da marginal dos dados e, neste caso, a comparação dos modelos se torna mais fácil, pois basta obter uma estimativa para a média ou para a média harmônica da verossimilhança caso a função de importância ($g(\theta_i)$) seja, respectivamente, a distribuição *a priori* e a distribuição *a posteriori* dos parâmetros.

As duas formas de estimar a marginal dos dados, apresentadas a seguir, são descritas segundo Sørensen & Gianola (2004). A distribuição marginal dos dados

pode ser expressa como

$$\begin{aligned}
 p(y|M_i) &= \frac{\int p(y|\theta_i, M_i)p(\theta_i|M_i)d\theta_i}{\int p(\theta_i|M_i)d\theta_i} \\
 &= \frac{\int p(y|\theta_i, M_i) \frac{p(\theta_i|M_i)}{g(\theta_i)} g(\theta_i) d\theta_i}{\int \frac{p(\theta_i|M_i)}{g(\theta_i)} g(\theta_i) d\theta_i} \quad (34)
 \end{aligned}$$

o denominador em (34) reescrito, como a seguir, pode ser interpretado como a esperança de $p(\theta_i|M_i)/g(\theta_i)$.

$$\int \frac{p(\theta_i|M_i)}{g(\theta_i)} g(\theta_i) d\theta_i = \lim_{m \rightarrow \infty} \left[\frac{1}{m} \sum_{j=1}^m \frac{p(\theta_i^{[j]}|M_i)}{g(\theta_i^{[j]})} \right] \quad (35)$$

em que $p(\theta_i^{[j]}|M_i)$ é a densidade a priori para o modelo i avaliada no j -ésimo valor amostrado para θ .

O numerador em (34) de forma análoga ao denominador pode ser escrito como:

$$\begin{aligned}
 &\int p(y|\theta_i, M_i) \frac{p(\theta_i|M_i)}{g(\theta_i)} g(\theta_i) d\theta_i \\
 &= \lim_{m \rightarrow \infty} \left[\frac{1}{m} \sum_{j=1}^m p(y|\theta_i^{[j]}, M_i) \frac{p(\theta_i^{[j]}|M_i)}{g(\theta_i^{[j]})} \right] \quad (36)
 \end{aligned}$$

em que $p(y|\theta_i^{[j]}, M_i)$ é avaliada no j -ésimo valor obtido da distribuição de importância.

Assim, fazendo $\omega = p(\theta_i^{[j]}|M_i) / g(\theta_i^{[j]})$, para m grande a distribuição marginal dos dados é estimada por meio da razão entre (36) e (35) como a seguir:

$$\hat{p}(y|M_i) = \frac{\sum_{j=1}^m \omega_i^{[j]} p(y|\theta_i^{[j]}, M_i)}{\sum_i^m \omega_i^{[j]}} \quad (37)$$

TABELA 3: Escala para interpretação do fator de Bayes.

B_{ij}	Evidência favorável a M_i
< 1	negativa (favorável a M_j)
de 1 a 3	duvidosa
de 3 a 10	substancial
de 10 a 30	forte
de 30 a 100	muito forte
> 100	decisiva

Fonte: Jeffreys (1961).

Se a função de importância for a distribuição a *priori*, então cada um dos pesos (ω_i) será igual a 1 e a distribuição marginal dos dados (37) resultará em:

$$\hat{p}(y|M_i) = \frac{1}{m} \sum_{j=1}^m p(y|\theta_i^{[j]}, M_i), \quad (38)$$

com $\theta_i^{[j]}$ retirado da distribuição a priori.

A vantagem do uso da distribuição a *priori* como função de importância é a facilidade de obtenção de uma amostra de θ e a desvantagem é que, em geral, com esses parâmetros, encontra-se valores muito pequenos para a $p(y|\theta)$.

Por outro lado, se a função de importância for a distribuição a *posteriori*, obtém-se maior probabilidade para os dados e, além disso, não há necessidade do conhecimento da forma desta distribuição, conforme a expressão em (40). A desvantagem é a instabilidade numérica para o estimador da distribuição marginal dos dados ($\hat{p}(y|M_i)$), mas esse problema pode ser contornado utilizando a escala logarítmica.

$$\begin{aligned} \omega &= \frac{p(\theta_i|M_i)}{p(\theta_i|y, M_i)} = \frac{p(\theta_i|M_i)}{\frac{p(y|\theta_i, M_i)p(\theta_i|M_i)}{p(y|M_i)}} \\ &= \frac{p(y|M_i)}{p(y|\theta_i, M_i)} \end{aligned} \quad (39)$$

e substituindo a expressão à direita da última igualdade em (37) obtêm-se:

$$\begin{aligned} \hat{p}(y|M_i) &= \frac{\sum_{j=1}^m \frac{p(y|M_i)}{p(y|\theta_i^{[j]}, M_i)} p(y|\theta_i^{[j]}, M_i)}{\sum_{j=1}^m \frac{p(y|M_i)}{p(y|\theta_i^{[j]}, M_i)}} \\ &= \frac{m}{\sum_{j=1}^m \frac{1}{p(y|\theta_i^{[j]}, M_i)}} = \left[\frac{1}{m} \sum_{j=1}^m \frac{1}{p(y|\theta_i^{[j]}, M_i)} \right]^{-1}. \end{aligned} \quad (40)$$

O estimador da distribuição marginal dos dados em (40) é a média harmônica dos valores da verossimilhança calculada em cada ciclo j do processo MCMC.

A interpretação do fator de Bayes segundo a escala dada por Jeffreys (1961) é apresentada na Tabela 3. Com esta escala é possível visualizar, dentre dois modelos em comparação, a intensidade com a qual um é superior ao outro.

3 MATERIAL E MÉTODOS

3.1 Material

Os dados (Tabela 4) utilizados para a ilustração da metodologia proposta neste trabalho foram obtidos pelo Centro Federal de Educação Tecnológica de Rio Verde (CEFET-RV) durante a realização da Exposição Agropecuária deste município no ano de 2006. A variável resposta consistiu de notas atribuídas por 36 provadores a amostras de banana da terra desidratada. Os tratamentos avaliados nesta análise foram três diferentes concentrações de sacarose utilizadas no processo de desidratação osmótico, sendo T1 - solução a 30%, T2 - solução a 40% e T3 - solução a 50% de sacarose.

Os provadores incluíam crianças e adultos de ambos os sexos, não treinados. Empregou-se a escala hedônica de nove pontos, cujo ponto 1 correspondeu a “desgostei muitíssimo” e o 9 a “gostei muitíssimo” (Dutcosky, 1996). A característica analisada foi a cor da banana da terra desidratada. O delineamento experimental adotado foi em blocos ao acaso em que cada provador avaliou os três tratamentos e, assim, este constituía um bloco completo.

3.2 Métodos

Dentre as distribuições que podem ser empregadas para explicar o fenômeno estudado, neste trabalho focou-se na comparação entre a *t* de Student e a normal.

Seguindo recomendação de Kizilkaya et al. (2002) as categorias foram reagrupadas de tal forma que em cada uma tivesse no mínimo cinco observações. O esquema de reorganização está descrito na Tabela 5.

TABELA 4: Notas atribuídas por 36 provadores à cor da banana da terra desidratada utilizando três concentrações de sacarose.

Provador	% de sacarose			Provador	% de sacarose		
	30	40	50		30	40	50
1	7	1	2	19	8	7	8
2	8	8	8	20	9	9	9
3	9	6	5	21	7	6	7
4	7	3	8	22	7	8	8
5	7	8	8	23	9	9	9
6	9	9	9	24	8	4	9
7	9	8	8	25	9	8	8
8	9	9	9	26	9	8	8
9	9	9	6	27	9	9	9
10	8	7	7	28	8	8	8
11	8	9	8	29	8	7	6
12	8	5	9	30	9	9	9
13	9	8	9	31	8	8	8
14	8	6	5	32	8	8	8
15	9	9	9	33	8	8	9
16	9	6	8	34	8	9	6
17	8	9	7	35	5	5	5
18	8	8	8	36	7	9	8

Fonte: Dados fornecidos pelo Centro Federal de Educação Tecnológica de Rio Verde - CEFET-RV.

TABELA 5: Esquema de reclassificação dos dados em cinco categorias.

Categorias	E. H. de 9 Pontos*	E. H. de 5 Pontos
Desgostei muitíssimo	1	
Desgostei muito	2	
Desgostei	3	1
Desgostei moderadamente	4	
Nem gostei nem desgostei	5	
Gostei moderadamente	6	2
Gostei	7	3
Gostei muito	8	4
Gostei muitíssimo	9	5

(*)Fonte: Dutcosky (1996).

A análise de variância (ANAVA) usual foi realizada para os dados sem transformação, com transformação logarítmica e com transformação de Box-Cox (Box & Cox, 1964). Também, foi apresentada a análise de variância para o modelo de regressão linear. A pressuposição de normalidade dos resíduos do modelo em cada transformação, foi estudada por meio do gráfico dos resíduos em função dos quantis teóricos (Normal Q-Q Plot) e, por meio do teste de Shapiro-Wilk (Shapiro & Wilk, 1965).

Na análise da variância, a variável resposta Y , foi modelada como segue:

$$Y_{ij} \sim N(\beta_i + u_j, \sigma^2),$$

em que Y_{ij} é a observação ou nota atribuída ao tratamento i ($i=1, 2, 3$), pelo provador (bloco) j ($j = 1, 2, \dots, 36$); β_i é a média do tratamento i , u_j o efeito do bloco j e σ^2 a variância residual.

Os elementos de Y são observados em uma das K categorias ordenadas e fazendo $p_{ik} = P[Y_i = k]$ pode-se definir as probabilidades acumuladas por meio da expressão $\eta_{ik} = \sum_{k=1}^{K-1} p_{ik}$. Segundo McCullagh (1980) um modelo para p_{ik} é dado por $\eta_{ik} = \Phi(\gamma_k - x'_i \theta)$, com $i = 1, 2, \dots, n$ e $k = 1, 2, \dots, K - 1$ e generalizando, $\eta_{ik} = G(\gamma_k - x'_i \theta)$. Portanto, verifica-se que a análise pode ser realizada por meio dos modelos lineares generalizados (GLM) cuja função de ligação é uma distribuição de probabilidade acumulada. Neste trabalho, foram atribuídas as distribuições normal e t de Student acumuladas para G .

Para a implementação do processo de Monte Carlo via Cadeias de Markov (MCMC), no caso em que se assumiu distribuição normal para a variável latente

L , o modelo para Y foi escrito como:

$$p(Y_i = k|\theta, \gamma) = \Phi\left(\frac{\gamma_k - w'_i\theta}{\delta^2}\right) - \Phi\left(\frac{\gamma_{k-1} - w'_i\theta}{\delta^2}\right)$$

em que $i = 1, 2, \dots, n$, com n representado o número de observações e $k = 1, 2, \dots, K$ é o índice para categorias, $\Phi(\cdot)$ a distribuição normal padrão acumulada e $\theta' = c(\beta', u')$. Introduzindo a variável latente L o modelo foi reescrito como segue:

$$p(Y_i = k|L_i, \gamma) = \sum_{k=1}^K I_{[\gamma_{k-1}; \gamma_k]}(L_i)I(Y_i = k)$$

com

$$L_i|\theta, \delta^2 \sim N(w'_i\theta, \delta^2)$$

Quando assumiu-se distribuição t de Student para L , a especificação do modelo foi como a seguir:

$$p(Y_i = k|\theta, \gamma) = F_\nu\left(\frac{\gamma_k - w'_i\theta}{\delta^2}\right) - F_\nu\left(\frac{\gamma_{k-1} - w'_i\theta}{\delta^2}\right)$$

em que F_ν representa a distribuição t de Student acumulada. O modelo em dois estágios resultou em:

$$p(Y_i = k|L_i, \gamma) = \sum_{k=1}^K I_{[\gamma_{k-1}; \gamma_k]}(L_i)I(Y_i = k)$$

com

$$L_i|\theta, \delta^2 \sim N(w'_i\theta, \frac{\delta^2}{\lambda_i})$$

e

$$p(\lambda_i|\nu) = \frac{(\frac{\nu}{2})^{\frac{\nu}{2}}}{\Gamma(\frac{\nu}{2})} \lambda_i^{\frac{\nu}{2}-1} \exp\left\{-\frac{\lambda_i}{2}\nu\right\} \quad (41)$$

Nos algoritmos AC e MC foi assumido, sem perda de generalidade, que $\delta^2 = 1$.

Foi considerada informação *a priori* vaga para β por meio da distribuição,

$$p(\beta) \propto \exp\left\{-\frac{0,001}{2}\beta'\beta\right\}$$

A especificação da *priori* para os efeitos aleatórios u foi feita por meio de uma distribuição normal multivariada como segue:

$$u|A \sim N(\emptyset, A\sigma_u^2),$$

em que \emptyset é um vetor nulo, de dimensão $q \times 1$. Em todas as abordagens consideradas neste trabalho, a matriz A é uma matriz identidade $q \times q$ por assumir que em experimento com blocos aleatorizados estes são não correlacionados.

As *prioris* para ν , δ^2 e σ_u^2 foram respectivamente:

$$p(\nu) \propto \frac{1}{(1+\nu)^2},$$

$$p(\sigma_u^2) \propto (\sigma_u^2)^{-(20+1)} \exp\left\{\frac{-5}{\sigma_u^2}\right\}, \quad (42)$$

e

$$p(\sigma_u^2) \propto (\sigma_u^2)^{-(3+1)} \exp\left\{\frac{-5}{\sigma_u^2}\right\} \quad (43)$$

Considerou-se as mesmas priors em todos os algoritmos meramente para fins de padronização eliminando, assim, a possibilidade de um algoritmo ser privilegiado por priors que contribuíssem para a convergência mais rápida ou prejudicado por priors que produzissem efeito contrário.

Os Três tratamentos do exemplo são níveis quantitativos igualmente espaçados do fator concentração de sacarose. Assim, o modelo de regressão linear simples pode ser ajustado com o modelo quadrático permitindo testar a falta de ajustado modelo linear.

As médias de tratamentos podem ser estruturadas em dois contrastes ortogonais referentes aos efeitos de regressão linear e quadrático, quais sejam:

- $C'_1\beta = \frac{\beta_3 - \beta_1}{2}$, associado à regressão linear para efeito de porcentagem de sacarose; e
- $C'_2\beta = \beta_2 - \frac{\beta_3 + \beta_1}{2}$, associado à regressão quadrática.

Tais contrastes podem ser diretamente calculados na amostra da distribuição a posteriori conjunta.

O coeficiente de correlação intraclassa (ρ),

$$\rho = \frac{\sigma_u^2}{\sigma_u^2 + \delta^2}$$

foi calculado para avaliar a semelhança entre as notas dos diferentes avaliadores.

As metodologias propostas por Sørensen et al. (1995) (seção 2.1) e por Kizilkaya et al. (2003) adaptada de Cowles (1996) (seções 2.2 e 2.3) foram aplicadas aos dados descritos anteriormente. Foram desenvolvidas, também, adaptações para modelos mistos, a partir da metodologia de Albert & Chib (1993) (seção 4.1.1) e

para modelos mistos com funções de ligação normal (seção 4.1.2) e t de Student (seção 4.1.2) acumuladas a partir do algoritmo de Nandram & Chen (1996).

Os algoritmos foram implementados em **R** (R Development Core Team, 2007) e permitem analisar quaisquer estruturas de covariância para os efeitos aleatórios.

Por meio da distribuição conjunta a *posteriori* em cada uma das situações obteve-se cadeias com 10000 iterações para cada parâmetro. Como um dos objetivos deste trabalho foi a verificação da convergência nos três algoritmos (AC, MC e NC) e nas duas funções de ligação utilizadas, para este fim, não foi considerada uma amostra válida para a realização de inferências e sim, uma cadeia de tamanho suficiente para o estudo da convergência em cada caso. Para a aplicação do critério de Gelman & Rubin foram consideradas duas cadeias para cada parâmetro.

Foram comparadas também a modelagem por meio da distribuição t de Student e por meio da distribuição normal. Nestes casos, foi necessário o uso da amostra válida, pois, além da velocidade de convergência dos algoritmos, o interesse também consistiu na verificação da melhor modelagem (distribuição t de Student ou normal). A escolha do melhor modelo foi realizada em cada um dos algoritmos, por meio do fator de Bayes, estimado por meio da média harmônica conforme a expressão em (40).

Para a realização das inferências foi considerada uma amostra final de tamanho 5000, obtida a partir de uma cadeia com 255000 iterações, em que as 5000 primeiras foram descartadas para a eliminação do efeito do valor inicial arbitrário e armazenado um valor a cada 50 para que estes fossem não correlacionados.

Essas cadeias foram utilizadas também para a estimação dos contrastes anteriormente citados.

A verificação da convergência foi realizada por meio de gráficos para as

cadeias dos parâmetros, pelos critérios de Raftery & Lewis (1992) e de Gelman & Rubin (1992).

Foi calculada a autocorrelação para todos os parâmetros no processo amostral. O cálculo dos valores de autocorrelações para diferentes intervalos (autocorrelação de *lag k*) entre os dados das cadeias de Markov foi feito como se segue:

$$r_k = \frac{E[(X_t - \bar{X})(X_{t+k} - \bar{X})]}{\sqrt{E[(X_t - \bar{X})^2]E[(X_{t+k} - \bar{X})^2]}}$$

em que X_t é o valor amostrado na iteração t , X_{t+k} o valor amostrado na iteração $t + k$ e \bar{X} é a média de todos os valores amostrados.

Estão apresentadas autocorrelações calculadas para os parâmetros correspondentes à média do tratamento e ao parâmetro *threshold* que apresentaram maior problema de autocorrelação em cada algoritmo.

4 RESULTADOS E DISCUSSÃO

4.1 Resultados metodológicos

Nesta seção são apresentados os desenvolvimentos das três adaptações aplicadas ao modelo misto, realizadas a partir das metodologias de Nandran & Chen (1996), com função de ligação normal e t de Student acumuladas e de Albert & Chib (1993) com função de ligação t de Student.

Na seção 4.2 será apresentado um exemplo de aplicação dos modelos de Albert & Chib (1993), Cowles (1996) e Nandran & Chen (1996), bem como suas adaptações na análise sensorial de alimentos. O estudo de convergência é realizado em cada caso.

4.1.1 Algoritmo adaptado de AC (AC-t)

A metodologia de Albert & Chib (1993) é adaptada neste caso, com a introdução de parâmetros de efeitos aleatórios na modelagem de L^* .

As distribuições condicionais completas para cada parâmetro foram obtidas a partir da equação (24) e são as que seguem:

$$\theta^* | L^*, \lambda, \nu \sim N \left(M^{-1} W' R^{-1} L^*, M^{-1} \right) \quad (44)$$

em que $M = W' R^{-1} W + V^{-1}$ e R é uma matriz diagonal com seus elementos dados por λ_i^{-1} $i = 1, 2, \dots, n$ e,

$$p(\lambda_i | \lambda_{-i}, L^*, \theta, \nu) \propto \lambda_i^{\left(\frac{\nu+1}{2}\right)-1} \exp \left(-\frac{\lambda_i}{2} \left((L_i^* - w_i' \theta)^2 + \nu \right) \right) \quad (45)$$

a qual é o núcleo de uma distribuição Gama com parâmetros dados a seguir e λ_{-i} ,

denota todos os elementos de λ exceto λ_i .

$$\lambda_i | \lambda_{-i}, L^*, \nu, \theta \sim G \left(\frac{\nu + 1}{2}, \frac{1}{2} \left((L_i^* - w_i' \theta)^2 + \nu \right) \right) \quad (46)$$

e

$$L_i^* | L_{-i}^*, \theta, \nu, \lambda_i \sim N \left(w_i' \theta^*, \frac{1}{\lambda_i} \right), \quad \gamma_{k-1}^* < l_i^* < \gamma_k^*, \quad k = 1, 2, 3, \dots, K. \quad (47)$$

A distribuição condicional completa a *posteriori* para ν não tem forma fechada e, assim, torna-se necessário o uso do algoritmo Metropolis-Hastings para a amostragem desta distribuição:

$$p(\nu | \theta^*, L^*, \gamma^*) \propto \left(\frac{(\frac{\nu}{2})^{(\nu/2)}}{\Gamma(\nu/2)} \right)^n \left(\prod_{i=1}^n \lambda_i^{\frac{\nu}{2}-1} \exp \left(-\frac{\nu}{2} \lambda_i \right) \right) \quad (48)$$

As distribuições condicionais para os parâmetros σ_u^2 e γ_k são, respectivamente, dadas pelas expressões (15) e (11).

A implementação do Amostrador de Gibbs pode ser feita amostrando seguidamente das distribuições condicionais completas, dadas em (44), (15), (46), (48), (47) e (11).

4.1.2 Algoritmo adaptado de NC - (NC-normal)

Nesta seção é apresentada a proposta de Nandram & Chen (1996) adaptada para o modelo misto, isto é, a variável latente é modelada em função de efeitos fixos e aleatórios. Estes autores propuseram a reparametrização conforme (49) e utilizaram a distribuição Dirichlet como candidata para os parâmetros de limiar,

$$\delta = 1/\gamma_{K-1}^*, \quad \gamma_k^{**} = \delta \gamma_k^*, \quad k = 0, 1, 2, \dots, K, \quad \theta^{**} = \delta \theta^* \text{ e } L^{**} = \delta L^*. \quad (49)$$

Desta forma, os parâmetros *threshold* continuam dividindo a reta em K intervalos e destes parâmetros, agora, existem apenas $K - 3$ desconhecidos. Observe que com essa reparametrização, para problemas com três categorias, não existirá nenhum parâmetro de limiar a ser estimado.

O Jacobiano da transformação é dado por $[\delta^2]^{-\frac{1}{2}(n+m+K)}$, em que n é o número de observações, m o número de variáveis explicativas e K a quantidade de categorias. Considerando uma distribuição a *priori* para δ^2 , a distribuição conjunta a *posteriori* em (8) fica sendo:

$$p(\theta^{**}, \gamma_k^{**}, \delta^2, L^{**} | y) \propto \left[\prod_{i=1}^n \phi(L_i^{**}, w_i \theta^{**}, \delta^2) I_{[\gamma_k^{**}, \gamma_{k+1}^{**})}(L_i^{**}) \right] \times \phi(\theta^{**}; \mathbf{0}, \delta^2 V) (\delta^2)^{-\frac{K}{2}} p(\sigma_u^2) p(\delta^2) \quad (50)$$

A distribuição condicional completa a *posteriori* para θ^{**} é,

$$\theta^{**} | L^{**}, \delta^2, Y, \sigma_u^2 \sim N \left(B^{-1} W' L^{**}, \delta^2 B^{-1} \right). \quad (51)$$

com B , neste caso, dado por: $B = \delta^2 V^{-1} + W' W$

Se $p(\delta^2)$ na expressão anterior for uma Gamma Inversa,

$$p(\delta^2) \propto (\delta^2)^{-(c+1)} \exp \left\{ -\frac{d}{\delta^2} \right\} \quad (52)$$

em que, c e d são os hiperparâmetros desta priori, então, a distribuição condicional completa a *posteriori* para δ^2 será:

$$\delta^2 | \theta^{**}, L^{**}, Y, \sigma_u^2 \sim GI(a_\delta, b_\delta) \quad (53)$$

com

$$a_\delta = \frac{(n + m + K + 2c)}{2}$$

e

$$b_\delta = \frac{\left((L^{**} - W\theta^{**})' (L^{**} - W\theta^{**}) + \theta^{**'} V^{-1} \theta^{**} + 2d \right)}{2}$$

A variância de efeitos aleatórios (σ_u^2) tem distribuição condicional como em (15).

Como já mencionado, a distribuição condicional de $(\gamma^{**}, L^{**} | \theta^{**}, \delta^2, Y)$ pode ser escrita como o produto das distribuições condicionais $(\gamma^{**} | \theta^{**}, \delta^2, Y)$ e $(L^{**} | \gamma^{**}, \theta^{**}, \delta^2, Y)$. E com a reparametrização em (49) a distribuição condicional para $(\gamma^{**} | \theta^{**}, \delta^2, Y)$ é

$$\begin{aligned} \pi(\gamma^{**} | \theta^{**}, \delta^2, Y) &\propto \prod_{Y_i=2} \left[\Phi \left(\frac{\gamma_2^{**} - w_i' \theta^{**}}{\delta} \right) - \Phi \left(\frac{-w_i' \theta^{**}}{\delta} \right) \right] \\ &\times \prod_{Y_i=3} \left[\Phi \left(\frac{\gamma_3^{**} - w_i' \theta^{**}}{\delta} \right) - \Phi \left(\frac{\gamma_2^{**} - w_i' \theta^{**}}{\delta} \right) \right] \dots \\ &\dots \times \prod_{Y_i=K-1} \left[\Phi \left(\frac{1 - w_i' \theta^{**}}{\delta} \right) - \Phi \left(\frac{\gamma_{K-2}^{**} - w_i' \theta^{**}}{\delta} \right) \right] \end{aligned} \quad (54)$$

e para $(L^{**} | \gamma^{**}, \theta^{**}, \delta^2, Y)$

$$L^{**} | \gamma^{**}, \theta^{**}, \delta^2, Y_i = k \sim N(W\theta^{**}, \delta^2). \quad (55)$$

Observe que para a amostragem de γ^{**} é necessário o uso do algoritmo

Metropolis-Hastings, pois este parâmetro não tem uma distribuição com forma na qual se possa aplicar o método de Monte Carlo. Cowles (1996) propôs a distribuição normal, para a geração de candidatos para um determinado *threshold*, truncada por outros dois adjacentes e Nandram & Chen (1996) propuseram a distribuição Dirichlet, a qual é uma distribuição conjugada da multinomial.

Uma motivação para o uso da distribuição Dirichlet é que os *threshold*, na forma reparametrizada, os valores de (γ_k^{**}) ficam no intervalo de zero a um, isto é, a partição em (2) resultará em:

$$(\gamma_0^{**}, \gamma_1^{**}); [\gamma_1^{**}, \gamma_2^{**}); \dots; [\gamma_{K-1}^{**}, \gamma_K^{**});$$

com $\gamma_1^{**} = 0$ e $\gamma_{K-1}^{**} = 1$ e assim, pode se construir uma variável p definida da seguinte forma:

$$p_{k-1} = \gamma_k^{**} - \gamma_{k-1}^{**}, \quad k = 2, \dots, K - 1;$$

em que

$$p = (p_1, p_2, \dots, p_{K-2})', \quad p_k \geq 0, \quad k = 1, 2, \dots, K - 2 \text{ e } \sum_{k=1}^{K-2} p_k = 1$$

De acordo com o teorema do valor médio,

$$\Phi \left(\frac{\gamma_k^{**} - w'_i \theta^{**}}{\delta} \right) - \Phi \left(\frac{\gamma_{k-1}^{**} - w'_i \theta^{**}}{\delta} \right) = \frac{1}{\delta} \phi \left(\frac{\xi_{k-1} - w'_i \theta^{**}}{\delta} \right) p_{k-1} \quad (56)$$

em que $\xi_{k-1} \in (\gamma_k^{**}; \gamma_{k-1}^{**})$, $k = 2, 3, \dots, K - 1$, e $\phi(\cdot)$ é a função densidade normal padrão.

Então, a partir de (56) tem-se:

$$\pi(\gamma^{**} | \theta^{**}, \delta^2, Y) \propto h_1(\xi) h_2(p) \quad (57)$$

e na expressão anterior, h_1 e h_2 são, respectivamente:

$$h_1(\xi) = \prod_{k=1}^{K-2} \prod_{i=1}^{n_k} \phi\left(\frac{\xi_k - w_i' \theta^{**}}{\delta}\right)$$

e

$$h_2(\mathbf{p}) = \prod_{k=1}^{K-2} p_k^{n_{k+1}} \quad (58)$$

Nesta adaptação, a distribuição Dirichlet também é usada como geradora de candidatos pois, a reparametrização dos parâmetros *threshold* é como desenvolvido por Nandram & Chen (1996).

Pode-se observar que a distribuição condicional de γ^{**} é dada pelo produto de h_1 por h_2 em que h_2 é o núcleo de uma distribuição Dirichlet com parâmetros $\mathbf{n} = (n_2 + 1, n_3 + 1, \dots, n_{K-1} + 1)'$ e h_2 não depende de θ^{**} e nem de δ .

Com os novos valores propostos para \mathbf{p} gerados a partir de h_2 a construção dos novos valores de γ_k^{**} é como segue:

$$\gamma_{k,j}^{**} = \sum_{i=1}^{k-1} p_{i,j}, \quad k = 2, 3, \dots, K - 2.$$

A probabilidade de aceitação do novo vetor

$$\gamma_j^{**} = \left(\gamma_{2,j}^{**}, \gamma_{3,j}^{**}, \dots, \gamma_{K-2,j}^{**} \right)$$

é o mínimo entre os valores um e α em que

$$\alpha = \omega(\gamma_j^{**}, \mathbf{p}_j) / \omega(\gamma^{**(j-1)}, \mathbf{p}^{j-1}), \quad (59)$$

e j representa a j -ésima iteração, do algoritmo. Em (59), a forma geral da distribuição de $\omega(\gamma^{**}, \mathbf{p})$ é dada por $\omega(\gamma^{**} | \theta^{**}, \delta^2) / p(\mathbf{p} | \mathbf{n}, \theta^{**}, \delta^2)$.

A expressão de $p(\gamma^{**}|\theta^{**}, \delta^2)$ é dada em (54) e $p(\mathbf{p}|\mathbf{n}, \theta^{**}, \delta^2)$ é a distribuição Dirichlet,

$$p(\mathbf{p}) = \frac{1}{Z(\mathbf{n})} \prod_{k=1}^{K-2} p_k^{n_{k+1}-1} \quad (60)$$

em que $p_1, \dots, p_{K-2} \geq 0$; $\sum_{k=1}^{K-2} p_k = 1$ e $n_2, \dots, n_{K-1} > 0$. Na expressão acima, $Z(\mathbf{n})$ é a constante normalizadora dada por

$$Z(\mathbf{n}) = \frac{\prod_{k=1}^{K-2} \Gamma(n_{k+1})}{\Gamma(\sum_{k=1}^{K-2} n_{k+1})}.$$

Como o interesse é na distribuição conjunta de L^{**} e γ^{**} , a amostragem é feita para \mathbf{p} e, a partir dos elementos deste vetor constrói-se γ_k^{**} , e caso o novo vetor de parâmetros *threshold* seja aceito atualiza-se os valores de L^{**} . Caso contrário continua com a amostra anterior da variável latente.

Para a implementação deste algoritmo, a amostragem é realizada a partir das expressões em (51), (15), (53), (60) e (55).

4.1.3 Algoritmo adaptado de NC - (NC-t)

Na forma reparametrizada, dada em (49), ou seja, de acordo com a metodologia apresentada por Nandram & Chen (1996) (algoritmo NC) e com a utilização da distribuição t de Student acumulada como função de ligação, a verossimilhança fica sendo:

$$P(Y_i = k|\theta^{**}, \delta, \nu, \gamma^{**}) = F_\nu \left(\frac{\gamma_k^{**} - w'_i \theta}{\delta} \right) - F_\nu \left(\frac{\gamma_{k-1}^{**} - w'_i \theta}{\delta} \right), \quad (61)$$

$k = 1, 2, \dots, K$

em que F_ν é a distribuição t-Student acumulada com ν graus de liberdade.

Como já visto, a distribuição t de Student pode ser escrita em dois estágios, os quais são dados agora, como segue:

$$L_i^{**} | \theta^{**}, \delta^2, \lambda_i \sim N \left(w_i' \theta^{**}, \frac{\delta^2}{\lambda_i} \right), i = 1, 2, \dots, n \quad (62)$$

$$\lambda_i | \nu \sim \text{Gamma} \left(\frac{\nu}{2}, \frac{\nu}{2} \right). \quad (63)$$

O modelo amostral em (61) será dado por:

$$P(Y_i = k | \theta^{**}, \delta, \lambda_i, \nu, \gamma^{**}) = \Phi \left(\frac{\gamma_k^{**} - w_i' \theta}{\frac{\delta}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{k-1}^{**} - w_i' \theta}{\frac{\delta}{\sqrt{\lambda_i}}} \right), \quad (64)$$

$k = 1, 2, \dots, K$

Para a obtenção da distribuição condicional completa própria para ν , Kizilkaya et al. (2003) propuseram a priori $p(\nu) = 1/(1 + \nu)^2$ e admitindo *priori* vaga para θ^{**} a *posteriori* conjunta de todos os parâmetros e a variável auxiliar ou latente é:

$$\begin{aligned} & p(\theta^{**}, \gamma_k^{**}, \delta^2, L^{**}, \lambda | y) \propto \\ & \propto (\delta^2)^{-\frac{K}{2}} \left[\prod_{i=1}^n \phi \left(L_i^{**}, w_i \theta^{**}, \frac{\delta^2}{\lambda_i} \right) I_{[\gamma_k^{**}, \gamma_{k+1}^{**})}(L_i^{**}) \right] \\ & \times \left[\prod_{i=1}^n \lambda_i^{\left(\frac{\nu}{2}\right)-1} \exp \left(-\frac{\lambda_i}{2} \nu \right) \right] \phi(\theta^{**}, 0, \delta^2 V) \frac{1}{(1 + \nu)^2} \\ & \times p(\sigma_u^2) p(\delta^2) p(\nu) \end{aligned} \quad (65)$$

em que $\lambda = \{\lambda_i\}_{i=1}^n$, V é como na seção 2.1, e as prioris para σ_u^2 e δ^2 são dadas como em (7) e (52), respectivamente.

As distribuições condicionais completas a *posteriori*, para cada parâmetro são dadas a seguir.

A distribuição condicional para θ , resulta em:

$$\theta^{**} | L^{**}, \delta^2, \nu, \lambda \sim N \left(M^{-1} W' R^{-1} L^{**}, \delta^2 M^{-1} \right) \quad (66)$$

em que $M = W' R^{-1} W + \delta^2 V^{-1}$.

O parâmetro λ_i tem distribuição condicional conforme expressão a seguir:

$$p(\lambda_i | \lambda_{-i}, L^{**}, \theta, \nu) \propto \lambda_i^{\left(\frac{\nu+1}{2}\right)-1} \exp \left(-\frac{\lambda_i}{2} \left((L_i^{**} - w_i' \theta)^2 + \nu \right) \right) \quad (67)$$

com λ_{-i} , denotando todos os elementos de λ exceto λ_i . A distribuição acima é proporcional a uma Gama com parâmetros $(\nu + 1)/2$ e $(L_i^{**} - w_i' \theta)^2 + \nu$, isto é,

$$\lambda_i | \lambda_{-i}, L_i^{**}, \theta, \nu, \delta^2 \sim G \left(\frac{\nu + 1}{2}, \frac{1}{2} \left((L_i^{**} - w_i' \theta)^2 + \nu \right) \right) \quad (68)$$

e

$$L_i^{**} | \lambda_i, \theta, \nu, \delta^2 \sim N \left(w_i' \theta^{**}, \frac{\delta^2}{\lambda_i} \right), \quad \gamma_{k-1} < l_i^{**} < \gamma_k, \quad k = 1, 2, 3, \dots, K \quad (69)$$

A distribuição condicional completa para ν não tem forma fechada e, assim, torna-se necessário o uso do algoritmo Metropolis-Hastings para a amostragem desta distribuição.

$$p(\nu | \theta^{**}, L^{**}, \gamma, \delta^2) \propto \left(\frac{\left(\frac{\nu}{2}\right)^{(\nu/2)}}{\Gamma(\nu/2)} \right)^n \left(\prod_{i=1}^n \lambda_i^{\frac{\nu}{2}-1} \exp \left(-\frac{\nu}{2} \lambda_i \right) \right) \frac{1}{(1+\nu)^2}. \quad (70)$$

A distribuição condicional completa para δ^2 tem a mesma forma como em (53).

Como para o caso do modelo normal, a distribuição conjunta de L^{**} e γ^{**} pode ser escrita por meio do produto das condicionais $(\gamma^{**} | \theta^{**}, \delta^2, \lambda, Y)$ e $(L^{**} | \gamma^{**}, \theta^{**}, \delta^2, \lambda, Y)$.

A distribuição condicional completa para a variável auxiliar é como em (69). Para γ^{**} a distribuição condicional completa é dada por:

$$\begin{aligned}
 p(\gamma^{**}|\theta^{**}, \delta^2, \lambda, Y) &\propto \prod_{Y_i=2} \left[\Phi \left(\frac{\gamma_2^{**} - w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{-w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) \right] \\
 &\times \prod_{Y_i=3} \left[\Phi \left(\frac{\gamma_3^{**} - w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_2^{**} - w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) \right] \\
 &\dots \prod_{Y_i=K-1} \left[\Phi \left(\frac{1 - w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{K-2}^{**} - w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) \right] \quad (71)
 \end{aligned}$$

E neste caso também, de acordo com o teorema do valor médio, têm-se:

$$\Phi \left(\frac{\gamma_k^{**} - x_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) - \Phi \left(\frac{\gamma_{k-1}^{**} - x_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) = \frac{1}{\frac{\delta}{\sqrt{\lambda_i}}} \phi \left(\frac{\xi_{k-1} - x_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) p_{k-1} \quad (72)$$

em que $\xi_{k-1} \in (\gamma_k^{**}; \gamma_{k-1}^{**})$, $k = 2, 3, \dots, K - 1$, e $\phi(\cdot)$ é a função densidade normal padrão.

E a partir de (72)

$$\pi(\gamma^{**}|\theta^{**}, \delta^2, Y) \propto h_3(\xi) h_4(p) \quad (73)$$

e em (71), h_3 e h_4 são, respectivamente:

$$\begin{aligned}
 h_3(\xi) &= \prod_{k=1}^{K-2} \prod_{i=1}^{n_k} \left(\frac{\xi_k - w_i' \theta^{**}}{\frac{\delta}{\sqrt{\lambda_i}}} \right) \\
 &e \\
 h_4(p) &= \prod_{k=1}^{K-2} p_k^{n_{k+1}} \quad (74)
 \end{aligned}$$

No caso de adotar a distribuição t de Student para a variável latente, os

procedimentos para gerar os candidatos, a construção dos *threshold* e a probabilidade de aceitação (59) são idênticos ao caso do modelo normal, com exceção do parâmetro de escala que será substituído por $\frac{\delta}{\sqrt{\lambda_i}}$ onde este for requerido.

Desta forma, o algoritmo pode ser implementado por meio de iterações sucessivas a partir de (66), (15), (53), (68), (70), (69) e da distribuição Dirichlet (60).

4.2 Resultados do ajuste para os três algoritmos

4.2.1 Algoritmo AC (t e normal)

Os resultados obtidos, conforme Figuras 2 e 3, mostram que as estimativas para os parâmetros de limiar, principalmente, possuem uma autocorrelação muito forte, dificultando o processo de convergência tanto com função de ligação normal quanto t de Student acumuladas. Esta análise é coerente com discussões apresentadas por Sørensen et al. (1995), Sørensen & Gianola (2004), Cowles (1996) e Nandram & Chen (1996), pois, estes autores destacaram o mesmo problema quando utilizaram o algoritmo AC.

Para os contrastes entre tratamentos com o uso desse algoritmo foi verificado que estes convergem rapidamente (Figura 11).

A representatividade das categorias quanto à quantidade de elementos tem relação direta com o processo de convergência. A redução do número de categorias de nove para cinco, conforme sugestão de Kizilkaya et al. (2003), aumentou a eficiência do processo de amostragem quanto à convergência. As análises com nove categorias não são aqui apresentadas.

As estimativas de médias de tratamentos, também, apresentaram forte autocorrelação entre seus valores (Figura 2), indicando a necessidade de uma cadeia longa para a obtenção de amostras adequadas para fins de inferências.

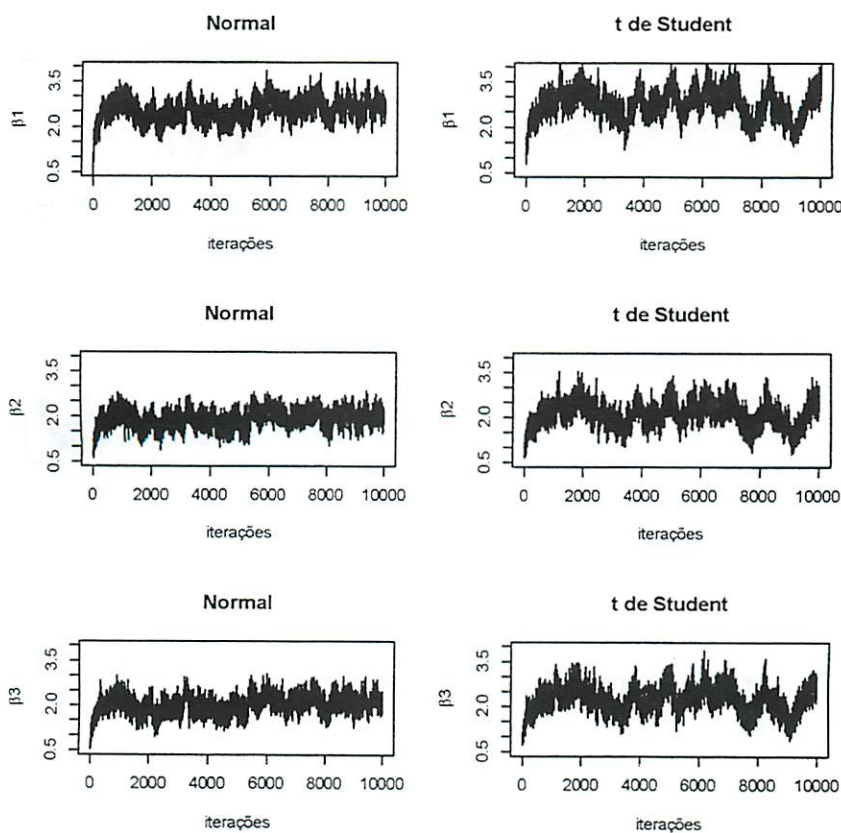


FIGURA 2: Cadeias para médias de tratamentos no algoritmo AC e nas duas funções de ligação normal de t de Student, acumuladas.

O número de iterações descartadas para eliminar o efeito do valor inicial arbitrário, isto é, o *Burn-in* (B) e o salto entre duas iterações consecutivas para que a amostra final obtida fosse considerada não correlacionada, ou seja, o *Thinning* (T) foi obtido por meio do critério de Raftery & Lewis (Tabela 6). Os maiores valores de B (descarte) e T (salto) foram, respectivamente, 248 e 31 para o parâmetro γ_2 , no modelo com função de ligação normal acumulada e 510 e 24 para γ_3 e σ_u^2 .

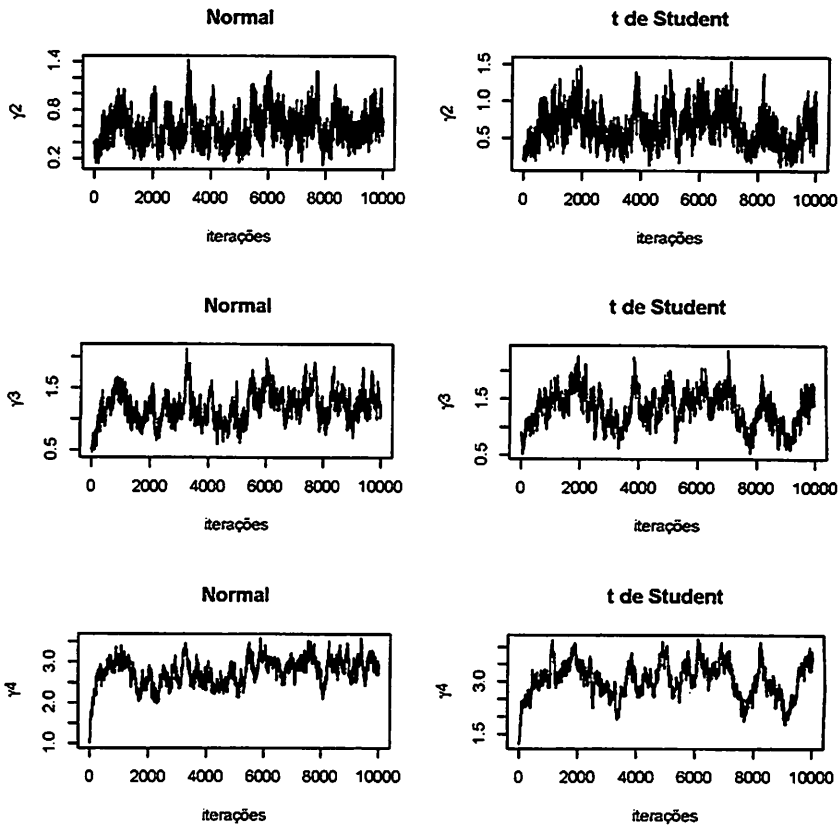


FIGURA 3: Cadeias para os parâmetros de limiar no algoritmo AC e modelos normal e t de Student.

no modelo com função de ligação t de Student acumulada.

Foram realizadas análises para os dados distribuídos em nove categorias em que os valores de B e T considerados foram, respectivamente, de 5000 e 100 em uma cadeia de 505000 e, portanto, obtendo uma amostra final de 5000 valores e, ainda assim, não obteve-se convergência. Nesta análise o algoritmo utilizado foi o AC com distribuição normal para a variável latente.

TABELA 6: Valores de *Burn-in* (B) e *Thinning* (T) obtidos por meio do diagnóstico de Raftery & Lewis, para as médias de tratamentos (β), variância de blocos (σ_u^2), variância residuais (δ^2) e graus de liberdade (ν), nos algoritmos AC, MC e NC segundo a distribuição da variável latente.

Parâmetro	AC				MC				NC			
	Normal		t		Normal		t		Normal		t	
	B	T	B	T	B	T	B	T	B	T	B	T
β_1	54	9	48	12	15	5	15	5	3	1	6	2
β_2	45	9	60	10	15	5	6	2	4	2	4	2
β_3	40	8	66	11	9	3	20	5	4	2	3	1
γ_2	45	9	50	10	30	10	128	16	4	2	4	2
γ_3	248	31	117	13	100	20	225	25	2	2	4	2
γ_4	209	19	510	10	85	17	252	28	-	-	-	-
σ_u^2	9	3	96	24	9	3	12	4	6	2	6	2
ν	-	-	20	5	-	-	72	18	-	-	78	3
δ_2	-	-	-	-	-	-	-	-	6	2	9	3

O maior fator de Brooks, Gelman & Rubin (\hat{R}) foi de 1,2584054 também para γ_2 no modelo normal e de 1,2233068 para γ_3 no modelo t de Student. Segundo Gelman & Rubin (1992), para que haja convergência o valor de \hat{R} deve estar próximo de 1. No entanto, segundo os mesmos autores, um valor próximo de 1,2 pode ser satisfatório para muitos casos. Se $\hat{R} \approx 1$, então o valor inicial arbitrário da cadeia não exerce mais efeito sobre as iterações (Paulino et al., 2003). É conveniente ressaltar que o valor \hat{R} é calculado usando a segunda metade de cada cadeia.

Em várias outras amostras obtidas esses valores apresentaram outra configuração, isto é, eles possuem oscilação. Porém com um valor de B de alguns poucos milhares de iterações e um valor de T de algumas poucas dezenas seria o suficiente para atender às exigências indicadas pelo critério de Raftery & Lewis em todas as situações.

O fato de considerar para a função G em (2) a distribuição t de Student

acumulada, não melhora a convergência do algoritmo em relação ao modelo normal (Figuras 2 e 3) e os valores de B , T e \hat{R} também, não diferem expressivamente do caso com função de ligação normal acumulada. Neste caso, o modelo é mais complexo, pois há acréscimo de parâmetros visto que a distribuição t de Student foi expressa por meio da mistura de uma normal e uma Gama para cada um dos parâmetros λ_i , associado às observações. A distribuição Gama para o parâmetro λ_i é dada em função do grau de liberdade ν , o qual não tem uma distribuição “fechada” para a implementação computacional, sendo necessário o uso do algoritmo Metropolis-Hastings para a amostragem de seus valores e isto contribui para a complexidade do algoritmo.

Os critérios de convergência aplicados e comentados anteriormente e principalmente os valores de B e T apontados pelo critério de Raftery & Lewis indicam a necessidade de cadeias pequenas em contraposição ao que se pode observar por meio das Figuras 2 e 3, pois verifica-se forte dependência entre as estimativas obtidas em cada iteração. Na Tabela 7 está ilustrada a dependência da média do tratamento e do parâmetro de limiar que apresentaram maior autocorrelação, tanto no caso em que a função de ligação foi a distribuição normal acumulada quanto no caso em que essa função foi a t de Student acumula, conforme Tabela 7. A justificativa para a apresentação da autocorrelação de apenas dois parâmetros se deve ao fato de que se a convergência não acontecer para apenas algumas das quantidades estimadas é suficiente para ilustrar a ineficiência do algoritmo.

Observa-se, ainda, que existe forte autocorrelação para esses parâmetros principalmente para *lag* 50 e *lag* 100 e no modelo t de Student verifica-se forte autocorrelação também no *lag* 200. Estes resultados são coerentes com as cadeias ilustradas nas Figuras 2 e 3.

Conforme valores de ν justifica-se o uso da distribuição t de Student neste

TABELA 7: Autocorrelação entre as estimativas para β_1 e para γ_3 no caso em que a função de ligação foi a distribuição normal acumulada e para β_2 e γ_2 quando a função de ligação foi a distribuição t de Student acumulada, no algoritmo AC.

Distribuição	Parâmetro	lag		
		50	100	200
Normal	β_1	0,1761	0,1039	0,0474
	γ_3	0,2264	0,1405	0,0637
t	β_2	0,3273	0,2451	0,1849
	γ_2	0,3420	0,2980	0,2330

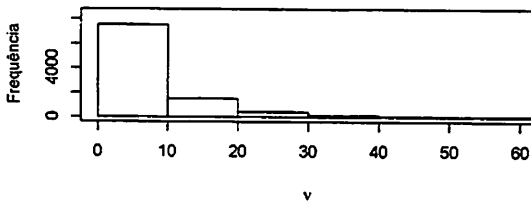
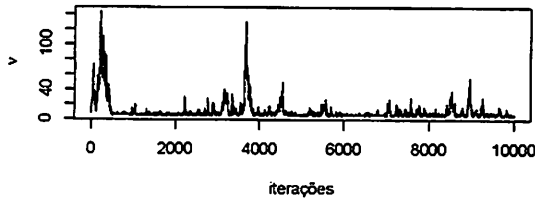


FIGURA 4: Cadeia e distribuição das estimativas de ν no algoritmo AC.

algoritmo (Figura 4), pois a maioria das estimativas foram abaixo de 10. No entanto, a dependência entre as estimativas para os parâmetros é maior neste caso. Este problema é solucionado adotando um grande valor de T.

Estudo de convergência na amostra válida para inferências, isto é, nas cadeias sem o efeito do valor inicial arbitrário e estimativas não autocorrelacionadas,

foi realizado por meio do critério de Gelman & Rubin. Este estudo, apresentou pequenas variações em torno do valor 1 e, portanto, considera-se que houve convergência, mesmo neste algoritmo, no qual foram verificadas dificuldades de convergência a partir do estudo inicial com 10000 iterações. Maiores valores de B e T dentre todas as cadeias foram, respectivamente, de 6 e 2 quando foi utilizada a distribuição normal padrão como função de ligação e de 16 e 5 quando se usou a distribuição t de Student na cadeia referente aos graus de liberdade (ν).

Para a escolha da melhor modelagem foi utilizado o quociente entre os valores obtidos a partir da expressão (40) com o uso da função de ligação, a distribuição normal padrão acumulada e da distribuição t de Student acumulada. A aproximação ao fator de Bayes, dada pela razão mencionada foi de 0,005862994 e tomando o inverso deste valor verifica-se, a partir da Tabela 3, que a evidência é decididamente favorável ao ajuste por meio da distribuição t de Student.

4.2.2 Algoritmo MC (t e normal)

Observando as Figuras, 2 e 5 e 3 e 6 verifica-se que não houve aumento da eficiência do algoritmo MC em relação ao AC. Por meio dos valores de B e T indicados para os parâmetros de limiar e apresentados a seguir, verifica-se que houve um pequeno aumento na eficiência do algoritmo MC. Os valores de autocorrelação para estes parâmetros indicam conclusões contrárias pois, em relação ao algoritmo AC possuem maior dependência entre as estimativas, como é ilustrado para γ_3 nas Tabelas 7 e 8.

Os maiores valores de B e T, no caso do uso da distribuição normal acumulada em (2) foram, respectivamente, 100 e 20 ambos, para o parâmetro γ_3 . Quando se usou a distribuição t de Student acumulada em (2), esses valores foram 252 e 28 para o parâmetro γ_4 (Tabela 6).

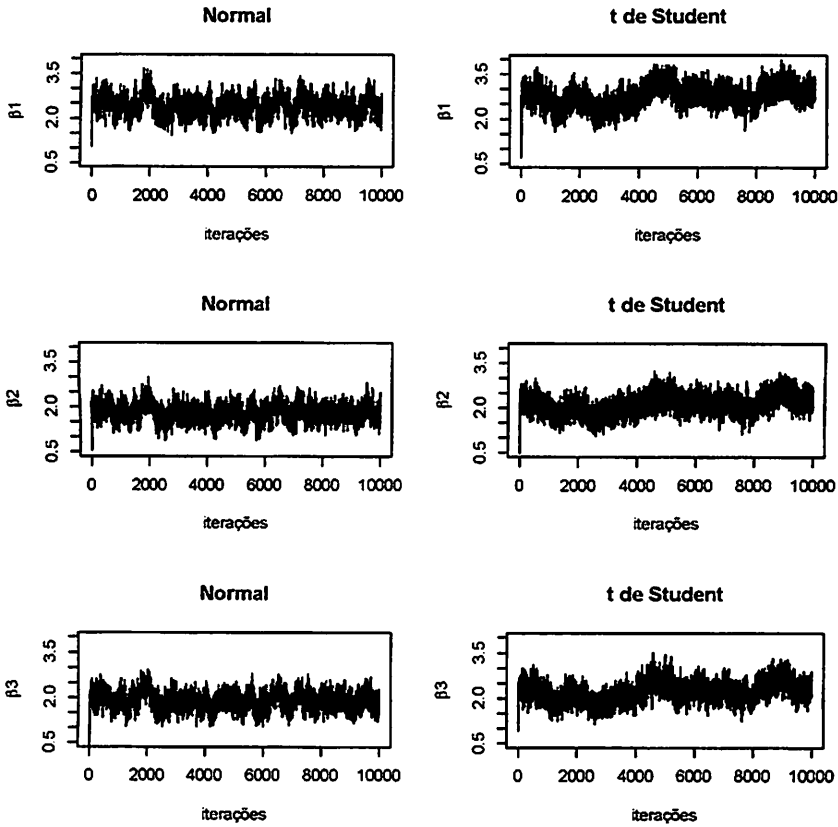


FIGURA 5: Cadeias para médias de tratamentos no algoritmo MC e nas duas funções de ligação normal de t acumuladas.

Por meio do valor de \hat{R} , o algoritmo MC apresenta melhor convergência em relação ao algoritmo AC com maior valor sendo 1,0685525 para parâmetro γ_2 no modelo normal e de 1,2272103 para os graus de liberdade (ν) no modelo t.

Assim como no algoritmo AC, a contradição entre os valores de B e T em relação à dependência pode ser verificada por meio dos valores destes e da autocorrelação indicada na Tabela 8.

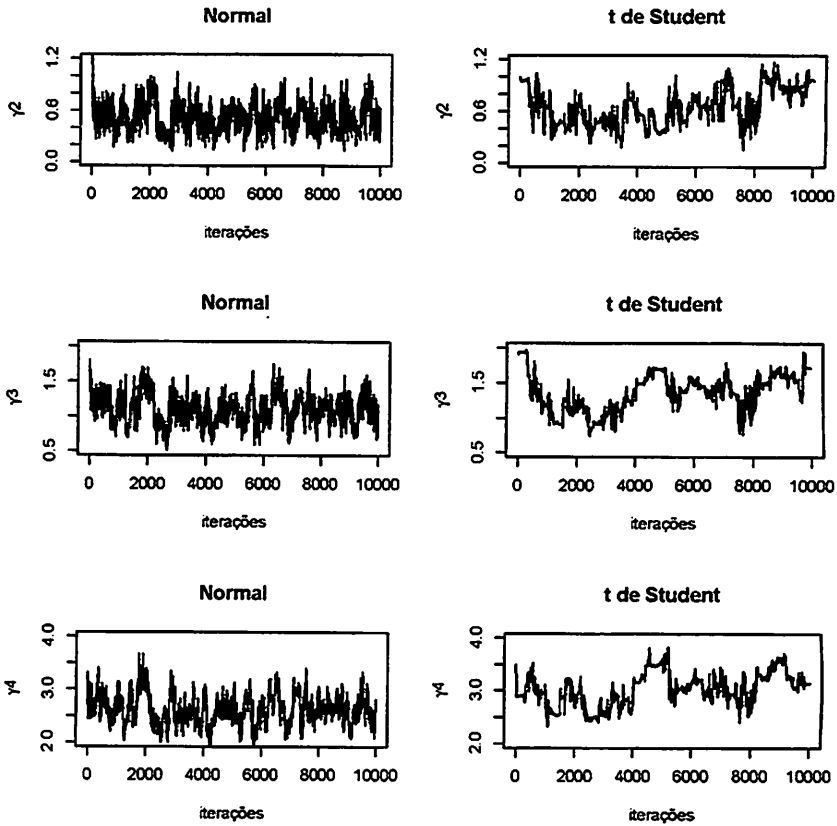


FIGURA 6: Cadeias para os parâmetros de limiar no algoritmo MC e modelos normal e t de Student.

Neste algoritmo também a redução de nove para cinco categorias melhorou a convergência nas situações em que foram usadas as distribuições normal e t acumuladas como funções de ligação.

Em relação ao melhor ajuste de modelos, com o uso das distribuições acumuladas normal e t de Student como funções de ligação, em (2), Kizilkaya et al. (2003) decidiram favoravelmente ao uso da segunda.

TABELA 8: Autocorrelação entre as estimativas de β_1 e de γ_4 no caso em que a função de ligação foi a distribuição normal acumulada e de β_1 e γ_3 quando foi usada a distribuição t de Student acumulada como função de ligação, no algoritmo MC.

Distribuição	Parâmetro	lag		
		50	100	200
Normal	β_1	0,3643	0,3276	0,0485
	γ_4	0,5453	0,4410	0,0596
t	β_1	0,5344	0,5009	0,4279
	γ_3	0,8777	0,8042	0,6790

Sabe-se que quando $\nu \rightarrow \infty$ a distribuição t de Student aproxima-se da distribuição normal padrão e, assim, se uma estimativa para ν é grande, não se justifica o uso da distribuição t acumulada em (2). Porém, neste algoritmo, as estimativas para ν , em sua grande maioria, foram estimadas abaixo de 20 o que pode ser verificado na Figura 7, mas para inferir sobre este parâmetro é necessário uma cadeia longa, haja vista a necessidade de considerar um intervalo grande entre duas iterações sucessivas para a obtenção de amostra final não correlacionada (Figura 7).

Neste algoritmo, os valores de \hat{R} calculados a partir da amostra válida para inferências, em todas as cadeias, ficaram dentro do intervalo aceitável para considerar convergência. Os maiores valores de B e T, também calculados na amostra final, foram, respectivamente, de 6 e 2 quando se usou a função de ligação normal padrão acumulada e de 78 e 3 para a cadeia relativa ao parâmetro ν quando se utilizou a distribuição t de Student (Tabela 6). Para os demais parâmetros, os valores de B e T foram similares nas duas modelagens. Portanto, por meio dos dois critérios verifica-se que houve convergência do algoritmo.

A contrastação dos dois modelos concorrentes, por meio da aproximação do fator de Bayes, indica que a decisão deve ser tomada com evidência substancial

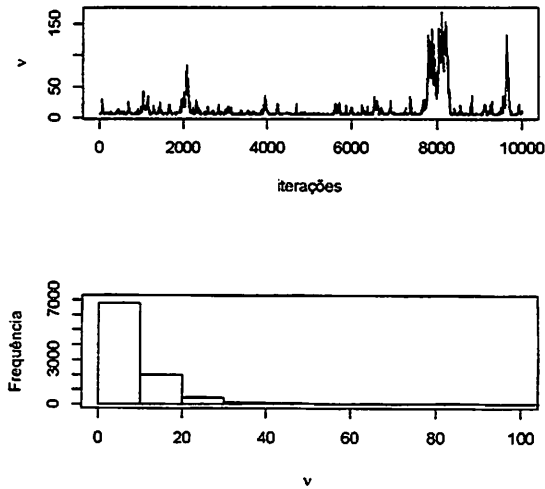


FIGURA 7: Distribuição das estimativas de ν no algoritmo MC.

(Tabela 3) a favor do uso da distribuição t de Student como função de ligação, já que a razão entre as médias harmônicas calculadas utilizando (40), no caso em que a função de ligação foi a normal padrão acumulada e no caso em que foi a distribuição t de Student, apresentou valor de 0,1862367.

4.2.3 Algoritmo NC (t e normal)

Em contraposição aos algoritmos propostos por Albert & Chib (1993) e por Cowles (1996) a reparametrização proposta por Nandram & Chen (1996) acelera expressivamente a convergência, diminuindo a autocorrelação e a necessidade do descarte de um grande número de valores iniciais para eliminar o efeito do valor inicial arbitrário.

Sob o aspecto da aceleração da convergência é indiferente o uso da dis-

tribuições acumuladas normal ou t de Student em (2), no algoritmo NC também. A principal diferença entre as duas abordagens é o acréscimo da quantidade de parâmetros a serem estimados quando se usa a distribuição t e como consequência tem-se maior complexidade na implementação do processo de amostragem.

Com este algoritmo é indiferente, também, a redução do número de categorias de nove para cinco. Exceto para o parâmetro ν , a convergência do algoritmo é rápida para todos os parâmetros.

A reparametrização NC consiste na multiplicação dos parâmetros e da variável latente por δ e, desta maneira, para retornar à escala original, basta dividi-los por este valor.

As Figuras 8 e 9 e os critérios adotados para o estudo de convergência, e comentados nos dois parágrafos seguintes, ilustram a vantagem do uso do algoritmo NC no modelo normal.

No caso do modelo normal, a indicação das maiores quantidades de B e T foram, respectivamente, 18 e 6 para o efeito do bloco 35 e de 150 e 25 para ν quando se utilizou a distribuição t de Student como função de ligação.

Segundo o critério de Gelman & Rubin, as cadeias para todos os parâmetros atingiram convergência, dado que o fator determinado por estes autores deve estar suficientemente próximo de 1 e isto ocorreu nesta análise exceto para o parâmetro ν na modelagem com função de ligação t de Student para G em (2).

Esses valores de descarte inicial (B) e pulo (T) são insignificantes quando comparados à valores que são geralmente adotados. Neste caso, mesmo adotando um comportamento conservador, e considerando quantidades maiores de valores para o descarte inicial e para o pulo, não teria uma cadeia tão grande como a adotada por Sörensen et al. (1995) com 620000 iterações, descarte inicial de 20000 e um valor tomado para a amostra válida a cada 20.

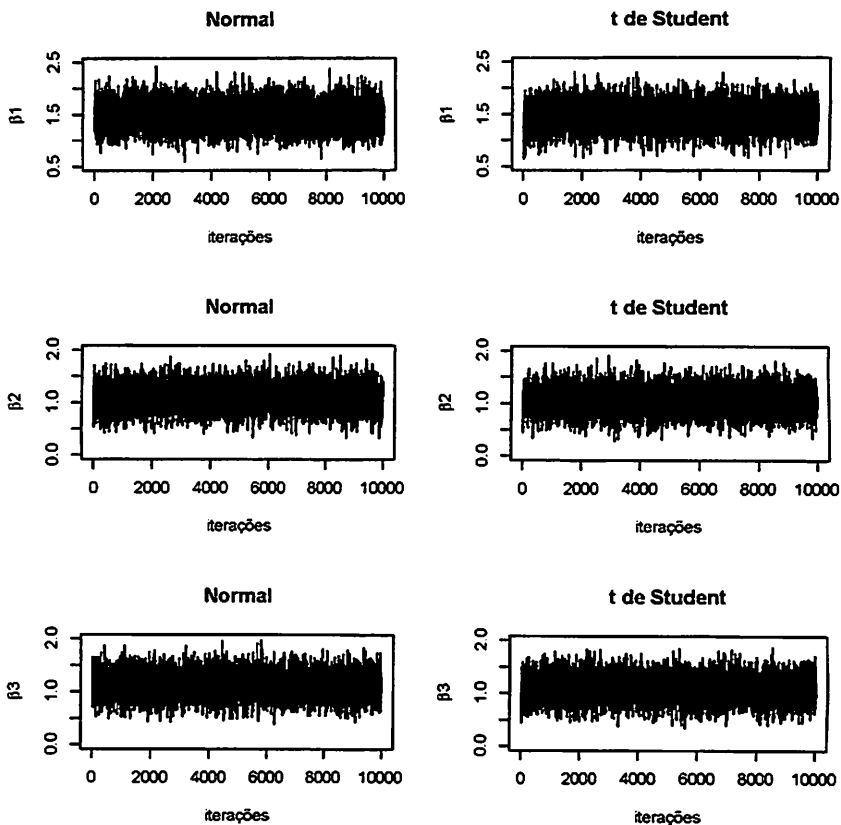


FIGURA 8: Cadeias para médias de tratamentos no algoritmo NC e nas duas funções de ligação normal de t de Student, acumuladas.

Ressalta-se que apesar dos números adotados por Sørensen et al. (1995) serem uma referência para comparação, a capacidade computacional nos dias atuais os tornam pequenos e, sendo assim, é útil a observação das cadeias e a constatação de que, a partir de poucas iterações, as amostras para um determinado parâmetro já são providas de uma mesma população com o algoritmo NC, e tanto

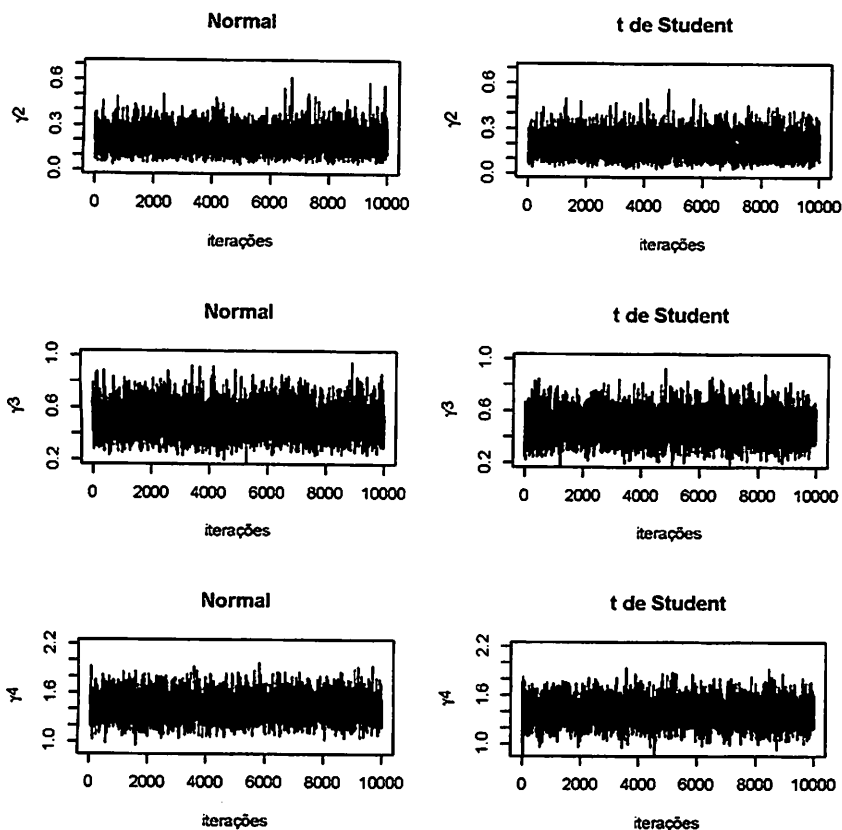


FIGURA 9: Cadeias para os parâmetros de limiar no algoritmo NC e modelos normal e t.

no modelo normal quanto no modelo t de Student. Além disto, as estimativas possuem pequena autocorrelação (Figuras 8 e 9). Em outras palavras, as cadeias se estabilizam rapidamente.

É notório a vantagem do algoritmo NC no que se refere à convergência. Esta afirmativa é confirmada por meio dos critérios utilizados e apontados aqui, além do comportamento de tendência e dependência das cadeias que podem ser

TABELA 9: Autocorrelação entre as estimativas de β_1 e de γ_2 no caso em que a função de ligação foi a distribuição normal acumulada e de β_1 e de γ_3 quando foi usada a distribuição t de Student acumulada como função de ligação, no algoritmo NC.

Distribuição	Parâmetro	lag		
		5	10	50
Normal	β_1	0,1198	0,0436	0,0003
	γ_2	0,0750	0,0197	0,0237
t	β_1	0,0962	0,0455	-0,0118
	γ_3	0,1274	0,0092	0,0068

observadas nas Figuras mencionadas.

Os valores de autocorrelação para os casos mais críticos de convergência, dentre as médias de tratamentos e dentre os parâmetros de limiar, são indicados na Tabela 9, assim como nos outros algoritmos. Porém, neste caso, foram considerados intervalos (*lags*) menores entre duas iterações para o cálculo da autocorrelação por considerar dispensáveis valores altos.

Os altos valores de ν observados por meio da Figura 10 implicam que a distribuição t de Student aproxima satisfatoriamente de uma distribuição normal padrão, o que permite escolher como função de ligação, neste algoritmo, a distribuição normal acumulada.

Porém, o parâmetro referente aos graus de liberdade (ν) apresenta problemas de convergências em todos os algoritmos e como em Kizilkaya et al. (2003), nesta análise também, houve problemas com relação à “mistura”, ou seja, não houve uma boa aleatorização dos valores amostrados (Figura 10), sendo necessário um intervalo grande entre duas iterações consecutivas amostradas para a obtenção de uma amostra final válida para a realização de inferências.

Outros parâmetros, para o algoritmo NC apresentaram rapidez de convergência similar aos citados na Tabela 9, indicando a necessidade de pequenos valo-

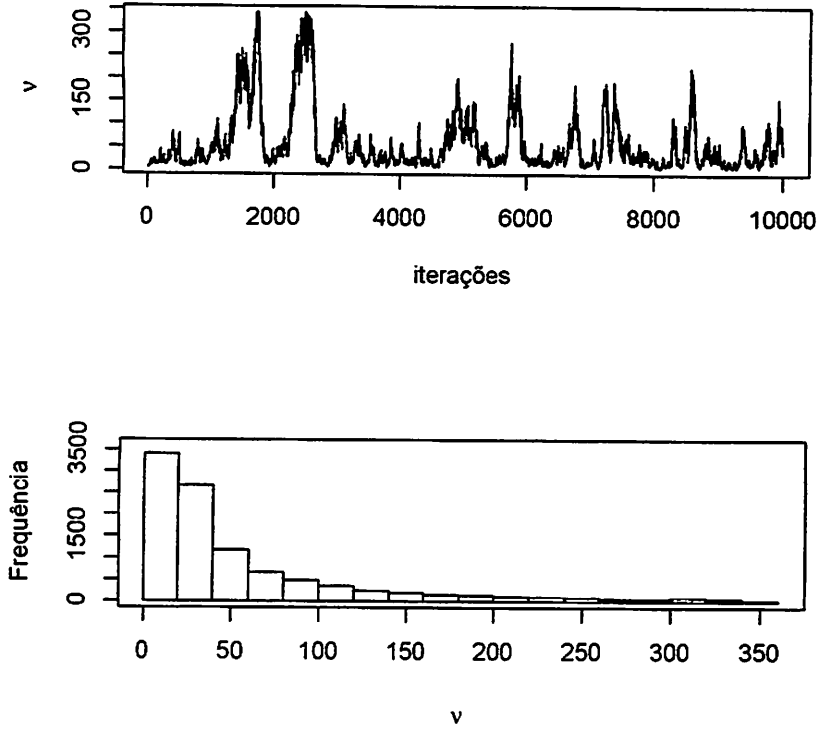


FIGURA 10: Cadeia e histograma para ν no algoritmo NC.

res de T e B.

Na amostra livre do efeito do valor inicial arbitrário e da autocorrelação encontrou-se valores de \hat{R} dentro do aceitável para considerar a convergência do algoritmo, exceto para a cadeia relativa à ν que foi de 1,3888694. No caso em que foi utilizada a distribuição normal padrão como função de ligação, o maior e menor valor de \hat{R} foram de 1,00167 e de 0,99980, ambos para efeito de blocos. Quando se usou a distribuição t de Student, o maior e menor valor de \hat{R} foram, res-

pectivamente, de 1,00238 e de 0,99983 para média do tratamento 3 (β_3) e efeito de bloco. Os maiores valores de B e T foram, respectivamente, 3 e 1 com a primeira função de ligação e 4 e 2 com a segunda.

Os valores estimados para os graus de liberdade são altos e indicam que a função de ligação pode ser a distribuição normal padrão acumulada e isto é confirmado por meio da aproximação do Fator de Bayes, calculada por meio de da expressão (40). Este critério permite concluir que a modelagem com o uso da distribuição normal padrão como função de ligação é ligeiramente superior à de Student com o valor da aproximação do fator de Bayes de 1,644553. Porém, segundo os critérios apresentados na Tabela 3, a evidência de que um modelo é melhor do que o outro, neste caso, é duvidosa.

Como destacado por Kizilkaya et al. (2003), inferências sobre ν não são confiáveis dada a dificuldade de convergência do algoritmo para este parâmetro. Neste trabalho, mesmo após a consideração do *Burn-in* e do *Thinning*, verificou-se que a cadeia para este parâmetro não é estável, com ocorrência de picos. Uma alternativa para amenizar este problema é o truncamento da distribuição deste parâmetro em 30, pois a distribuição t de Student com graus de liberdade acima deste valor aproxima-se da distribuição normal padrão.

No algoritmo NC é indiferente o uso da distribuição normal ou t de Student para a variável latente, o que foi observado a partir do fator de Bayes para os modelos com essas duas distribuições. Neste algoritmo, a distribuição geradora de candidatos é a distribuição Dirichlet, a qual, por sua característica, faz com que os parâmetros *threshold* fiquem limitados no intervalo $[0, 1]$ e isto modifica a escala da variável latente, fazendo com que os dois modelos tenham caudas semelhantes.

4.2.4 Análise dos contrastes de interesse

Os resultados de contrastes, bem como da correlação intraclasse (ρ) são apresentados a seguir. Nos algoritmos AC e MC a variância residual foi considerada fixa e igual a 1 ($\delta^2 = 1$).

Em caso de similaridade dessas estatísticas pode se decidir pela equivalência dos métodos quanto à inferências. Porém, ressalta-se que inferências confiáveis devem ser realizadas a partir de cadeias estáveis, ou seja, daquelas em que houve convergência e, por conseqüência, amostras da distribuição condicional conjunta a *posteriori* representam uma boa aproximação das distribuições marginais dos parâmetros.

Na Tabela 10 estão listados os HPDs em todas as situações estudadas para o contraste associado à regressão linear ($C_1'\theta$), associado ao desvio de regressão ($C_2'\theta$) e à correlação intraclasse (ρ).

É possível verificar por meio da Tabela 10 que os resultados são similares em todos os algoritmos, independente da função de ligação utilizada. No estudo inicial, com 10000 iterações, apenas o algoritmo NC apresentou estabilidade em suas cadeias, porém com os valores de B e T adotados para a obtenção da amostra válida para inferências obteve-se convergência em todos os casos. Os HPDs e as médias apresentadas na Tabela 10 foram calculados a partir da amostra válida.

Apesar da dificuldade de convergência dos algoritmos AC e MC e, principalmente para o primeiro, a convergência dos contrastes é evidente. Ilustração desta situação é apresentada na Figura 11. Os valores de B e T, iguais a 6 e 2, respectivamente, foram suficientes para eliminar o efeito do valor inicial arbitrário e para a obtenção de uma amostra final não autocorrelacionada. A maior autocorrelação foi detectada para o efeito da regressão linear com função de ligação t de Student acumulada com 0,4411 no *lag* 1 e 0,0894 no *lag* 5, indicando que em todos

TABELA 10: HPDs para funções dos parâmetros associadas à regressão linear ($C_1\beta$), ao desvio de regressão ($C_2\beta$) e à correlação intraclassa (ρ).

Algoritmo	Distribuição	Contraste	HPD	Média
AC	Normal	$C_1\beta$	[-0,5208; 0,0212]	-0,2581
	t	$C_1\beta$	[-0,6716; 0,0534]	-0,2981
	Normal	$C_2\beta$	[-0,8134; 0,1346]	-0,3355
	t	$C_2\beta$	[-1,0329; 0,2129]	-0,4190
	Normal	ρ	[0,4515; 0,7632]	0,6029
	t	ρ	[0,5195; 0,8522]	0,6921
MC	Normal	$C_1\beta$	[-0,5473; 0,0156]	-0,2562
	t	$C_1\beta$	[-0,6470; 0,0454]	-0,2887
	Normal	$C_2\beta$	[-0,8214; 0,1389]	-0,3345
	t	$C_2\beta$	[-1,0243; 0,2252]	-0,4035
	Normal	ρ	[0,4447; 0,7458]	0,5997
	t	ρ	[0,5045; 0,8416]	0,6788
NC	Normal	$C_1\beta$	[-0,6470; 0,0454]	-0,2887
	t	$C_1\beta$	[-0,2615; 0,0851]	-0,0866
	Normal	$C_2\beta$	[-1,0243; 0,2252]	-0,4034
	t	$C_2\beta$	[-0,4212; 0,1829]	-0,1208
	Normal	ρ	[0,5168; 0,7755]	0,6439
	t	ρ	[0,5100; 0,7698]	0,6425

os contrastes a convergência é obtida com um número pequeno de iterações.

4.3 Comparações com a ANAVA e considerações gerais

A análise de variância usual, para os dados sem transformação e com transformação logarítmica, indica que a porcentagem de sacarose influencia significativamente a nota atribuída à cor da banana da terra desidratada (Tabela 11) o que não ocorreu quando foi aplicada a transformação de Box-Cox. Foi detectado efeito linear significativo por meio do teste F, a 5% de probabilidade, apenas no caso da transformação logarítmica.

O poder da análise reduziu, no caso em que foi usada a transformação de Box-Cox, pois, nesta análise não foi detectado influência significativa dos trata-

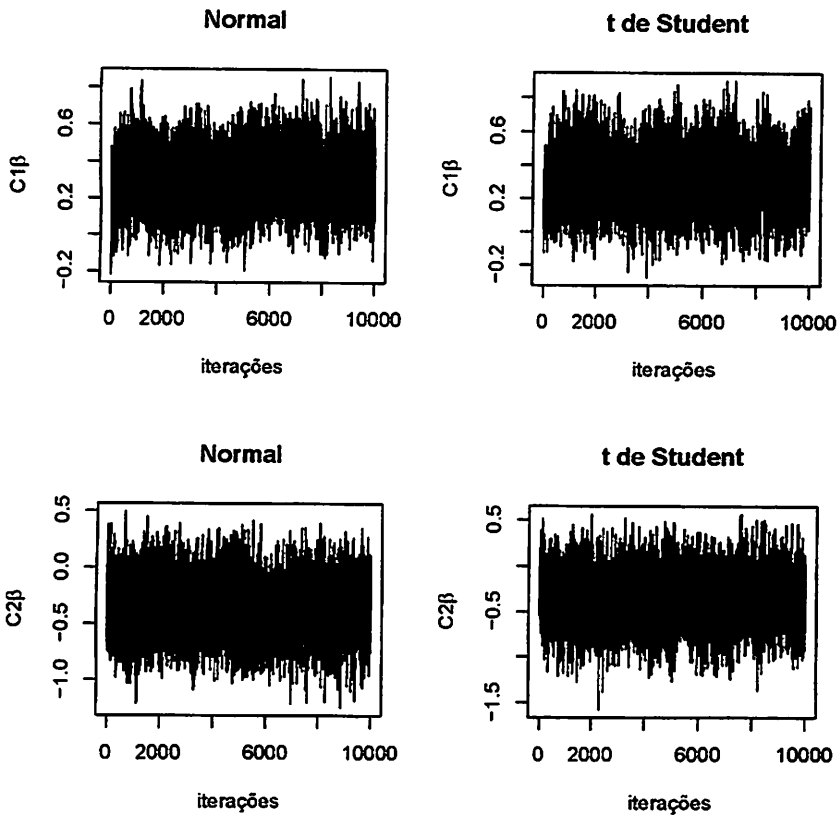


FIGURA 11: Cadeias para os contrastes associados à regressão linear e ao desvio de regressão no algoritmo AC e funções de ligação normal e t de Student, acumuladas.

mentos na variável cor (Tabela 11). O valor de λ para a transformação de Box-Cox foi de 3,5.

Em todos os três casos, conforme valor p da Tabela 11, a falta de ajuste foi não significativo. Isto é coerente com o contraste $C_2\beta$ estimado a partir das cadeias para a média de cada tratamento pois, o HPD para este contraste não inclui o valor zero e pode-se concluir então que o desvio de regressão é não significativo.

TABELA 11: Quadro-resumo da análise da variância para a variável cor sem transformação, com transformação logarítmica e com transformação Box-Cox.

Fonte Variação	GL	Tipo de transformação					
		Sem		Logarítmica		Box-Cox	
		<i>F</i>	valor p	<i>F</i>	valor p	<i>F</i>	valor p
Bloco	35	3,18	0,0000	3,19	0,0000	3,18	0,0000
Tratamento	2	3,19	0,0471	3,80	0,0272	2,38	0,0999
Efeito linear	1	3,58	0,0626	4,06	0,0476	2,86	0,0954
Desvio de regressão	1	2,81	0,0984	3,53	0,0644	1,90	0,1721

A hipótese de normalidade dos resíduos, segundo o teste de Shapiro-Wilk não foi aceita em nenhum dos casos. As probabilidades de significância foram respectivamente 0,0004421, 0,000007276 e 0,001082 nas análises com os dados sem transformação, com transformação logarítmica e com transformação de Box-Cox. A ausência de normalidade dos resíduos pode ser observada também, na Figura 12.

Em resumo, nenhuma das análise de variâncias pode ser considerada válida, mas a mais próxima de ser válida (usando a transformação Box-Cox) apresenta conclusões semelhantes às dos algoritmos propostos.

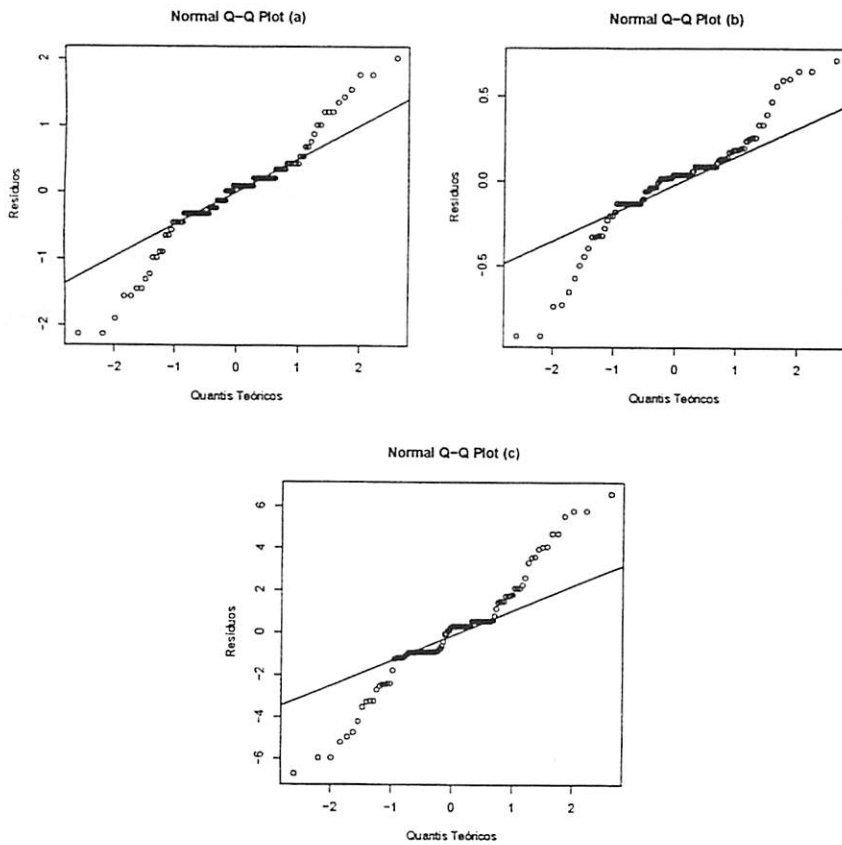


FIGURA 12: Gráfico de resíduos a partir do modelo ajustado para a variável nota atribuída à cor da banana da terra desidratada, sem transformação (a), com transformação logarítmica (b) e com transformação de Box-Cox (c).

5 CONCLUSÕES

- As adaptações para modelos mistos do algoritmo NC com funções de ligação normal padrão e t de Student são as mais eficientes e podem ser recomendadas.
- A distribuição t de Student é em geral uma função de ligação mais flexível, mesmo que no exemplo para o algoritmo NC seja indiferente o seu uso em alternativa à distribuição normal.
- A análise bayesiana de parâmetros de limiar é recomendável na modelagem de dados discretos ordinais provenientes da análise sensorial de alimentos.

REFERÊNCIAS BIBLIOGRÁFICAS

ALBERT, J. H.; CHIB, S. Bayesian analysis of binary and polychotomous response data. **Journal of the American Statistical Association**, Washington, v. 88, n. 422, p. 669-679, June 1993.

BOX, G.E.P; COX, D.R. An analysis of transformations (with discussion). **Journal of the Royal Statistical Society, Series B**, London, v. 26, n. 2, p. 211-252, Apr. 1964.

CHIB, S. Markov chain Monte Carlo methods: Computation and Inference. In: HECKMAN, J. J. and LEAMER E. (Eds.). **Handbook of Econometrics: Volume 5**. North-Holland: Elsevier Science, 2001. p. 3569-3649.

COWLES, M. K. Accelerating Monte Carlo Markov chain convergence for cumulative link generalized linear models, **Statistics and Computing**, Netherlands: Springer, v.6, n. 2 p. 101-111, June 1996.

DUTCOSKY, S. D. **Análise sensorial de alimentos**, Curitiba:Champagnat, 1996. 123p.

GELFAND, A.E.; SMITH A.F.M. Sampling-based approaches to calculating marginal densities, **Journal of the American Statistical Association**, Washington, v. 85, n. 410, p. 398-409, June 1990.

GELMAN, A.; CARLIN, J. B.; STERN, H. S.; RUBIN, D. B. **Bayesian data analysis**. London: Chapman and Hall, 2003. 668 p.

GELMAN, A. and RUBIN, D. B. Inference from iterative simulation using multiple sequences. **Statistical Science**, Hayward, v. 7, n. 4, p. 457-72, Nov. 1992.

HASTINGS, W. K. Monte Carlo sampling methods using Markov chains and their applications. **Biometrika**, London, v. 57, n. 1, p. 97-109, Apr. 1970.

JEFFREYS, H. **Theory of probability**. Oxford: Claredon Press, 1961. 470 p.

KEMPTHORNE, O.; **An introduction to genetic statistic**. New York: John Wiley and Sons Inc., 1966. 545 p.

ANEXO

PROGRAMAS	Páginas
PROGRAMA 1	Rotina R para implementação do algoritmo AC no caso em que a variável latente tem distribuição normal. 71
PROGRAMA 2	Rotina R para implementação do algoritmo AC no caso em que a variável latente tem distribuição t de Student. 75
PROGRAMA 3	Rotina R para implementação do algoritmo MC no caso em que a variável latente tem distribuição normal. 79
PROGRAMA 4	Rotina R para implementação do algoritmo MC no caso em que a variável latente tem distribuição t de Student. 84
PROGRAMA 5	Rotina R para implementação do algoritmo NC no caso em que a variável latente tem distribuição normal. 90
PROGRAMA 6	Rotina R para implementação do algoritmo NC no caso em que a variável latente tem distribuição t de Student. 95

SÖRENSEN, D.A.; ANDERSEN S.; GIANOLA D.; KORSGAARD I. Bayesian inference in threshold models using Gibbs sampling, **Genetics Selection Evolution**, Les Ulis, v. 27, n. 3, p. 229-249, 1995.

SÖRENSEN, D.; GIANOLA, D. **Likelihood, bayesian and MCMC methods in quantitative genetics**. United States of America: Springer, 2004. 740 p.

WRIGHT, S.; An analysis of variability in number of digits in an inbred strain of guinea pigs, **Genetics**, Pittsburgh, v. 19, p. 506-536, Nov. 1934.

ANEXO

PROGRAMAS	Páginas
PROGRAMA 1	Rotina R para implementação do algoritmo AC no caso em que a variável latente tem distribuição normal. 71
PROGRAMA 2	Rotina R para implementação do algoritmo AC no caso em que a variável latente tem distribuição t de Student. 75
PROGRAMA 3	Rotina R para implementação do algoritmo MC no caso em que a variável latente tem distribuição normal. 79
PROGRAMA 4	Rotina R para implementação do algoritmo MC no caso em que a variável latente tem distribuição t de Student. 84
PROGRAMA 5	Rotina R para implementação do algoritmo NC no caso em que a variável latente tem distribuição normal. 90
PROGRAMA 6	Rotina R para implementação do algoritmo NC no caso em que a variável latente tem distribuição t de Student. 95

Programa 1. Rotina R para implementação do algoritmo AC no caso em que a variável latente tem distribuição normal.

```
#####
## Algoritmo AC #
## Variável latente com distribuição normal #
#####

dados <- read.table("dados.txt",h=T)
attach(dados)

##### Definição do tamanho da cadeia de saída #

amostra <- 5000
burn <- 5000
jump <- 50
iter <- jump*amostra+burn

#####

library(VGAM)
library(MASS)
library(mvtnorm)

y <- cor
n <- length(y)

#### reorganização dos dados em 5 categorias ##

for(i in 1:n)
{
  if (y[i]<=5) y[i]<-1
}
for(i in 1:n)
{
  if (y[i] == 6) y[i]<- 2
}
for(i in 1:n)
{
  if (y[i] == 7) y[i]<- 3
}
for(i in 1:n)
{
  if (y[i] == 8) y[i]<- 4
}
for(i in 1:n)
{
  if (y[i] == 9) y[i]<- 5
}

## Matrizes de incidência de tratamentos e blocos #

trat <- c(rep(1,36),rep(2,36),rep(3,36))
bloco <- dados$bloco

XT <-matrix(0,n,3)
```

```

for(i in 1:n)
{
  XT[i,trat[i]]<-1
}

XB <-matrix(0,n,35)
for(i in 1:n)
{
  ifelse(bloco[i]==36, XB[i,] <- -i, XB[i,bloco[i]] <- 1)
}
W <- cbind(XT,XB)

##### Valores iniciais arbitrários #####

L      <- c(1:n)
for(i in 1:n)
{
  L[i]<- qnorm(pnorm(0)+pnorm(0)*runif(1))
}

theta  <- ginv(t(W)%*%W)%*%t(W)%*%L
e      <- L - W%*%theta
A      <- diag(35)      ## constante
ve     <- 1            ## constante
vu     <- 3
taunovo <- c(0, 0.2, 0.7, 1.1)
tau    <- taunovo
cont   <- 0
delta2 <- 1

#### objetos necessários para o cálculo da verossimilhança

vero   <- rep(1,n)
verossim <- NULL

##### prioris #####

se     <- 1000
ru     <- 3
su     <- 5

##### Amostrador de Gibbs #####

for (i in 1:iter)
{
##### condicional para theta #####

V      <- rbind(cbind(se*diag(3)      , matrix(0,3,35)),
               cbind(matrix(0,35,3), vu*A))
S      <- ginv(V)
theta  <- ginv(t(W)%*%W + S)%*%t(W)%*%L
var    <- ve*ginv(t(W)%*%W+ S)
thetaest <- rmvnorm(1,mean = theta,sigma = var, method = "chol")
theta  <- t(thetaest)
beta   <- theta[1:3]
u      <- theta[4:38]

```

```

##### condicional para variancia de blocos #####

c1 <- (35+2*ru)/2
c2 <- (t(u)%*%ginv(A)%*%u + 2*su)/2
vu <- rgamma(1,c1,c2)
vu <- as.real(1/vu)

##### condicional para L #####

m <- W%*%theta

for(k in 1:n)
{
  if(y[k]==1)
  {
    L[k] <- qnorm(pnorm(tau[1],m[k],ve) +
                 runif(1,mean=m[k],sd=sqrt(ve))
    vero[k] <- pnorm((tau[1] - m[k])/sqrt(ve))
  }
  if(y[k]==2)
  {
    L[k] <- qnorm(pnorm(tau[1],m[k],ve) + (pnorm(tau[2],m[k],ve) -
        pnorm(tau[1],m[k],ve)) * runif(1,mean=m[k],sd=sqrt(ve))
    vero[k] <- pnorm((tau[2] - m[k])/sqrt(ve)) -
        pnorm((tau[1] - m[k])/sqrt(ve))
  }
  if(y[k]==3)
  {
    L[k] <- qnorm(pnorm(tau[2],m[k],ve) + (pnorm(tau[3],m[k],ve) -
        pnorm(tau[2],m[k],ve)) * runif(1,mean=m[k],sd=sqrt(ve))
    vero[k] <- pnorm((tau[3] - m[k])/sqrt(ve)) -
        pnorm((tau[2] - m[k])/sqrt(ve))
  }
  if(y[k]==4)
  {
    L[k] <- qnorm(pnorm(tau[3],m[k],ve) + (pnorm(tau[4],m[k],ve) -
        pnorm(tau[3],m[k],ve)) * runif(1,mean=m[k],sd=sqrt(ve))
    vero[k] <- pnorm((tau[4] - m[k])/sqrt(ve)) -
        pnorm((tau[3] - m[k])/sqrt(ve))
  }
  if(y[k]==5)
  {
    L[k] <- qnorm(pnorm(tau[4],m[k],ve) +
        (1-pnorm(tau[4],m[k],ve)) * runif(1,mean=m[k],sd=sqrt(ve))
    vero[k] <- (1 - pnorm(- m[k]/sqrt(ve)))
  }
}

##### Cálculo da verossimilhança #####

verossim <- prod(vero)

```

```
##### condicional para os par. de limiar #####
min <- as.vector(tapply(L,y,min))
max <- as.vector(tapply(L,y,max))

for (j in 2:4)
{
  tau[j] <-runif(1,(max[j]),(min[j+1]))
}

##### Gera a saída no arquivo cadeia.txt #####

saida <- cbind(t(theta),t(tau),vu,verossim)
colunas <- length(saida)

if(i > burn && (i-burn)%jump == 0)
{
  write(saida, file="cadeia.txt", ncol=colunas, append=TRUE)
}

}
```

Programa 2. Rotina R para implementação do algoritmo AC no caso em que a variável latente tem distribuição t de Student.

```
#####
## Algoritmo AC #
## Variável latente com distribuição t #
#####

dados <-read.table("dados.txt",h=T)
attach(dados)

##### Definição do tamanho da cadeia de saída #

amostra <- 5000
burn <- 5000
jump <- 50
iter <- jump*amostra+burn

#####

library(VGAM)
library(MASS)
library(mvtnorm)
y <- dados$cor
n <- length(y)

#### reorganização dos dados em 5 categorias ##

for(i in 1:n)
{
  if (y[i]<=5) y[i]<-1
}
for(i in 1:n)
{
  if (y[i] == 6) y[i]<- 2
}
for(i in 1:n)
{
  if (y[i] == 7) y[i]<- 3
}
for(i in 1:n)
{
  if (y[i] == 8) y[i]<- 4
}
for(i in 1:n)
{
  if (y[i] == 9) y[i]<- 5
}

## Matrizes de incidência de tratamentos e blocos #

trat <- c(rep(1,36),rep(2,36),rep(3,36))
bloco <- dados$bloco

XT <-matrix(0,108,3)
for(i in 1:108)
```



```

    {
      XT[i, trat[i]] <- -1
    }

XB <- matrix(0, 108, 35)
for(i in 1:108)
  {
    ifelse(bloco[i]==36, XB[i,] <- -1, XB[i, bloco[i]] <- 1)
  }
W <- cbind(XT, XB)

##### Valores iniciais arbitrários #####

L <- c(1:n)
for(i in 1:n)
  {
    L[i] <- qnorm(pnorm(0)+pnorm(0)*runif(1))
  }
theta <- ginv(t(W)%*%W)%*%t(W)%*%L
e <- L - W%*%theta
A <- diag(35) ## constante
nu <- 10
vu <- 0.5
ve <- 1 ## constante
taunovo <- c(0, 0.4, 1.0, 1.4)
tau <- taunovo
pnuc <- 0.5
pnu <- 0.5
lambda <- c(rep(1, n))
R <- diag(n)
### objetos necessários para o cálculo da verossimilhança

vero <- rep(1, n)
verossim <- NULL

##### prioris #####

se <- 1000
ru <- 3
su <- 5

##### Amostrador Gibbs #####

for (i in 1:iter)
  {

##### condicional para theta #####

V <- rbind(cbind(se*diag(3) , matrix(0,3,35)),
           cbind(matrix(0,35,3), vu*A))
S <- ve*ginv(V)
theta <- ginv(t(W)%*%R%-%W + S)%*%t(W)%*%R%*%L
var <- ve*ginv(t(W)%*%R%*%W+ S)
thetaest <- rmvnorm(1, mean = theta, sigma = var, method = "chol")
theta <- t(thetaest)
beta <- theta[1:3]

```

```

u      <- theta[4:38]

##### condicional para variancia de blocos #####

c1 <- (35+2*ru)/2
c2 <- (t(u)*%ginv(A)%*%u + 2*su)/2
vu <- rgamma(1,c1,c2)
vu <- as.real(1/vu)

##### Amostra lambda (GS elemento a elemento) ###

for (j in 1:n)
{
  lambda[j] <- rgamma(1,(nu+1)/2,(nu+(L[j]-W[j,])%*%theta)^2)/2
}
R <- diag(lambda)

##### Amostra nu (MH) #####

nuc <- 2
while(nuc < 3)
{
  nuc <- rpois(1,nu)
}
pnu <- n*((nu/2)*log(nu/2)-lgamma(nu/2))+sum(((nu/2)-1)*
  log(lambda) + (-(nu/2)*lambda)) - 2*log(1+nu)
pnuc <- n*((nuc/2)*log(nuc/2)-lgamma(nuc/2))+sum(((nuc/2)-1)*
  log(lambda) + (-(nuc/2)*lambda)) - 2*log(1+nuc)
pnu <- exp(pnu)
pnuc <- exp(pnuc)

dnu <- dpois(nu,nu)
dnuc <- dpois(nuc,nu)

num <- pnuc*dnu
denom <- pnu*dnuc

ifelse((num > denom),(alfa <- 1),(alfa <- as.real(num/denom)))

teste <- runif(1)
if(teste < alfa)
{
  nu <- nuc
}

##### condicional para L #####

m <- W%*%theta
dp <- rep(1,n)
dp <- sqrt(ve/lambda)

for(k in 1:n)
{
  if(y[k]==1)
  {

```

```

L[k] <- qnorm(pnorm(tau[1],m[k],sd=dp[k])*
             runif(1,mean=m[k],sd=dp[k]))
vero[k] <- pnorm((tau[1] - m[k])/dp[k])
}
if (y[k]==2)
{
L[k] <- qnorm(pnorm(tau[1],m[k],sd=dp[k])+(pnorm(tau[2],m[k],sd=dp[k])
      -pnorm(tau[1],m[k],sd=dp[k]))*runif(1,mean=m[k],sd=dp[k]))
vero[k] <- pnorm((tau[2] - m[k])/dp[k]) - pnorm((tau[1] - m[k])/dp[k])
}
if (y[k]==3)
{
L[k] <- qnorm(pnorm(tau[2],m[k],sd=dp[k])+(pnorm(tau[3],m[k],sd=dp[k])
      -pnorm(tau[2],m[k],sd=dp[k]))*runif(1,mean=m[k],sd=dp[k]))
vero[k] <- pnorm((tau[3] - m[k])/dp[k]) - pnorm((tau[2] - m[k])/dp[k])
}
if (y[k]==4)
{
L[k] <- qnorm(pnorm(tau[3],m[k],sd=dp[k])+(pnorm(tau[4],m[k],sd=dp[k])
      -pnorm(tau[3],m[k],sd=dp[k]))*runif(1,mean=m[k],sd=dp[k]))
vero[k] <- pnorm((tau[4] - m[k])/dp[k]) - pnorm((tau[3] - m[k])/dp[k])
}
if (y[k]==5)
{
L[k] <- qnorm(pnorm(tau[4],m[k],sd=dp[k]) +
             (1-pnorm(tau[4],m[k],sd=dp[k]))*runif(1,mean=m[k],sd=dp[k]))
vero[k] <- (1 - pnorm(- m[k]/dp[k]))
}
}

##### Cálculo da verossimilhança #####

verossim <- prod(vero)

##### Amostragem dos thresholds #####

min <- as.vector(tapply(L,y,min))
max <- as.vector(tapply(L,y,max))
for(j in 2:4)
{
tau[j] <-runif(1,(max[j]),(min[j+1]))
}

##### Gera a saída no arquivo cadeia.txt #####

saida <- cbind(t(theta),t(tau),nu,vu,verossim)
colunas <- length(saida)

if(i > burn && (i-burn)%jump == 0)
{
write(saida, file="cadeia.txt", ncol=colunas, append=TRUE)
}
}

```

Programa 3. Rotina R para implementação do algoritmo MC no caso em que a variável latente tem distribuição normal.

```
#####
## Algoritmo MC #
## Variável latente com distribuição normal #
#####

dados <- read.table("dados.txt",h=T)
attach(dados)

##### Definição do tamanho da cadeia de saída #

amostra <- 5000
burn <- 5000
jump <- 50
iter <- jump*amostra+burn

#####

library(VGAM)
library(MASS)
library(mvtnorm)

y <- cor
n <- length(y)

#### reorganização dos dados em 5 categorias ##

for(i in 1:n)
{
  if (y[i]<=5) y[i]<-1
}
for(i in 1:n)
{
  if (y[i] == 6) y[i]<- 2
}
for(i in 1:n)
{
  if (y[i] == 7) y[i]<- 3
}
for(i in 1:n)
{
  if (y[i] == 8) y[i]<- 4
}
for(i in 1:n)
{
  if (y[i] == 9) y[i]<- 5
}

## Matrizes de incidência de tratamentos e blocos #

trat <- c(rep(1,36),rep(2,36),rep(3,36))
bloco <- dados$bloco
XT <-matrix(0,n,3)
for(i in 1:n)
```

```

{
  XT[i,trat[i]]<-1
}
XB <-matrix(0,n,35)
for(i in 1:n)
{
  ifelse(bloco[i]==36, XB[i,] <- -1, XB[i,bloco[i]] <- 1)
}
W <- cbind(XT,XB)

## objetos para o cálculo da verossimilhança ##

vero      <- rep(1,n)
verossim  <- NULL

##### Valores iniciais arbitrários #####

ve      <- 1
vu      <- 3
tau     <- c( 0, 0.2, 0.7, 1.0, 1000)
taunovo <- c( 0, 0.1, 0.9, 1.2, 1000)
L       <- c(1:n)
for(i in 1:n)
{
  L[i]<- qnorm(pnorm(0)+pnorm(0)*runif(1))
}

##### Número de categorias #####
K      <- 5

##### desvio padrão para a candidata ###
dp     <- 0.15
d      <- rep(1,iter) # vetor para atualização do dp

##### prioris #####

se     <- 1000
ru     <- 3
su     <- 5

##### Inicia a rotina do amostrador Gibbs ###

for (i in 1:iter)
{
  m     <- W%*%theta

##### condicional para theta #####

V      <- rbind(cbind(se+diag(3)      , matrix(0,3,35)),
               cbind(matrix(0,35,3), vu*A))
S      <- ve+ginv(V)
theta  <- ginv(t(W)%*%W + S)%*%t(W)%*%L
var    <- ve+ginv(t(W)%*%W+ S)

```

```

thetaest <- rmvnorm(1,mean = theta,sigma = var, method = "chol")
theta    <- t(thetaest)
beta     <- theta[1:3]
u        <- theta[4:38]

##### condicional para variancia de blocos###

c1 <- (35+2*ru)/2
c2 <- (t(u)%*%ginv(A)%*%u + 2*su)/2
vu <- rgamma(1,c1,c2)
vu <- as.real(1/vu)

##### Amostragem dos thresholds #####

for(j in 2:(K-1))
{
  areal    <- pnorm(taunovo[j-1],mean=tau[j],sd=dp)
  area2    <- pnorm(tau[j+1],mean=tau[j],sd=dp)-
             pnorm(taunovo[j-1],mean=tau[j],sd=dp)
  area     <- areal+runif(1)*area2
  taunovo[j] <- qnorm(area,mean=tau[j],sd=dp)
}

p1<-rep(0,K-2)
p2<-rep(0,K-2)
p3<-rep(0,K-2)
p4<-rep(0,K-2)

for(j in 1:(K-2))
{
  p1[j] <- pnorm((tau[j+2] -tau[j+1])/dp)
}
for(j in 1:(K-2))
{
  p2[j] <- pnorm((taunovo[j] -tau[j+1])/dp)
}
for(j in 1:(K-2))
{
  p3[j] <- pnorm((taunovo[j+2] -taunovo[j+1])/dp)
}
for(j in 1:(K-2))
{
  p4[j] <- pnorm((tau[j] -taunovo[j+1])/dp)
}

for(k in 1:n)
{
  if(y[k]==1)
  {
    ftau[k]    <- pnorm(tau[1] - m[k])
    ftaunovo[k] <- pnorm(taunovo[1] - m[k])
  }
  if(y[k]==2)
  {
    ftau[k]    <- pnorm(tau[2] - m[k]) - pnorm(tau[1] - m[k])
    ftaunovo[k] <- pnorm(taunovo[2] - m[k]) - pnorm(taunovo[1] - m[k])
  }
}

```

```

    }
  if (y[k]==3)
  {
    ftau[k]      <- pnorm(tau[3] - m[k]) - pnorm(tau[2] - m[k])
    ftaunovo[k] <- pnorm(taunovo[3] - m[k]) - pnorm(taunovo[2] - m[k])
  }
  if (y[k]==4)
  {
    ftau[k]      <- pnorm(tau[4] - m[k]) - pnorm(tau[3] - m[k])
    ftaunovo[k] <- pnorm(taunovo[4] - m[k]) - pnorm(taunovo[3] - m[k])
  }
  if (y[k]==5)
  {
    ftau[k]      <- 1 - pnorm(tau[4] - m[k])
    ftaunovo[k] <- 1 - pnorm(taunovo[4] - m[k])
  }
}

alfa <- exp(sum(log((p1-p2))-log((p3-p4)))+sum(log(ftaunovo)-log(ftau)))
teste <- runif(1)
R <- min(1,alfa)

if(teste < R)
{
  tau <- taunovo
##### conditional para L #####
for(k in 1:n)
{
  if(y[k]==1)
  {
    L[k]      <-qnorm(pnorm(tau[1],m[k])*runif(1),mean=m[k])
    vero[k] <-pnorm((tau[1] - m[k])/sqrt(ve))
  }
  if(y[k]==2)
  {
    L[k]      <-qnorm(pnorm(tau[1],mean=m[k])+(pnorm(tau[2],mean=m[k])
      -pnorm(tau[1],mean=m[k]))*runif(1),mean=m[k])
    vero[k] <-pnorm((tau[2] - m[k])/sqrt(ve)) -
      pnorm((tau[1] - m[k])/sqrt(ve))
  }
  if(y[k]==3)
  {
    L[k]      <-qnorm(pnorm(tau[2],mean=m[k])+(pnorm(tau[3],mean=m[k])
      -pnorm(tau[2],mean=m[k]))*runif(1),mean=m[k])
    vero[k] <-pnorm((tau[3] - m[k])/sqrt(ve)) -
      pnorm((tau[2] - m[k])/sqrt(ve))
  }
  if(y[k]==4)
  {
    L[k]      <-qnorm(pnorm(tau[3],mean=m[k])+(pnorm(tau[4],mean=m[k])
      -pnorm(tau[3],mean=m[k]))*runif(1),mean=m[k])
    vero[k] <-pnorm((tau[4] - m[k])/sqrt(ve)) -
      pnorm((tau[3] - m[k])/sqrt(ve))
  }
}

```

```

if(y[k]==5)
{
  L[k]    <-qnorm(pnorm(tau[4],mean=m[k]) +
                (1-pnorm(tau[4],mean=m[k]))*runif(1),mean=m[k])
  vero[k] <-(1 - pnorm( - m[k]/sqrt(ve)))
}
}
}

##### Cálculo da verossimilhança #####

verossim <- prod(vero)

##### atualização do desvio padrão da candidata #####

d[i] <- c(tau[2])
ifelse(i > 20, dp <- sd(d[(i-20):i]), dp <- 0.25)

##### Gera a saída no arquivo cadeia.txt #####

saida <- cbind(t(theta),t(tau),vu,dp,verossim)
colunas <- length(saida)

if(i > burn && (i-burn)%jump == 0)
{
  write(saida, file="cadeia.txt", ncol=colunas, append=TRUE)
}
}

```


Programa 4. Rotina R para implementação do algoritmo MC no caso em que a variável latente tem distribuição t de Student.

```
#####
## Algoritmo MC #
## Variável latente com distribuição t #
#####

dados<-read.table("dados.txt",h=T)

##### Definição do tamanho da cadeia de saída #

amostra <- 5000
burn <- 5000
jump <- 50
iter <- jump*amostra+burn

#####

library(VGAM)
library(MASS)
library(mvtnorm)
attach(dados)
y <- cor
n <- length(y)

### reorganização dos dados em 5 categorias ##

for(i in 1:n)
{
  if (y[i]<=5) y[i]<-1
}
for(i in 1:n)
{
  if (y[i] == 6) y[i]<- 2
}
for(i in 1:n)
{
  if (y[i] == 7) y[i]<- 3
}
for(i in 1:n)
{
  if (y[i] == 8) y[i]<- 4
}
for(i in 1:n)
{
  if (y[i] == 9) y[i]<- 5
}

## Matrizes de incidência de tratamentos e blocos #

trat <- c(rep(1,36),rep(2,36),rep(3,36))
XT <- matrix(0,108,3)
for(i in 1:108)
{
```

```

    XT[i,trat[i]]<-1
  }
XB <- matrix(0,108,35)
for(i in 1:108)
{
  ifelse(bloco[i]==36, XB[i,] <- -1, XB[i,bloco[i]] <- 1)
}
W <- cbind(XT,XB)

##### Valores iniciais arbitrários #####

theta <- ginv(t(W)%*%W)%*%t(W)%*%L
e <- L - W%*%theta
A <- diag(35)
ftau <- rep(1,n)
ftaunovo <- ftau
vu <- 3
ve <- 1
tau <- c( 0, 0.3, 0.9, 1.5, 1000)
taunovo <- c( 0, 0.6, 0.8, 2.0, 1000)
nu <- 10
pnu <- 0.5
pnuc <- 0.5
lambda <- c(rep(1,n))
R <- diag(n)
L <- c(1:n)
for(i in 1:n)
{
  L[i]<- qnorm(pnorm(0)+pnorm(0)*runif(1))
}

##### Número de categorias #####
K <- 5

##### desvio padrão para a candidata ###
dpc <- 0.15
dc <- rep(1,iter) # vetor para atualização do dpc

##### prioris #####
se <- 1000
ru <- 3
su <- 5

## objetos para o cálculo da verossimilhança ##

vero <- rep(1,n)
verossim <- NULL

##### Inicia a rotina do amostrador Gibbs ###

for (i in 1:iter)
{
  m <- W%*%theta

```

```

##### condicional para beta ###

V      <- rbind(cbind(se=diag(3)      , matrix(0,3,35)),
                cbind(matrix(0,35,3), vu*A))
S      <- ve*ginv(V)
theta  <- ginv(t(W)%*%R%*%W + S)%*%t(W)%*%R%*%L
var    <- ve*ginv(t(W)%*%R%*%W+ S)
thetaest <- rmvnorm(1,mean = theta,sigma = var, method = "chol")
theta  <- t(thetaest)
beta   <- theta[1:3]
u      <- theta[4:38]

### condicional para variancia de blocos ###

c1 <- (35+2*ru)/2
c2 <- (t(u)%*%ginv(A)%*%u + 2*su)/2
vu <- rgamma(1,c1,c2)
vu <- as.real(1/vu)

##### Amostragem dos thresholds #####

for(j in 2:(K-1))
{
  area1 <- pnorm(taunovo[j-1],mean=tau[j],sd=dpc)
  area2 <- pnorm(tau[j+1],mean=tau[j],sd=dpc)-
          pnorm(taunovo[j-1],mean=tau[j],sd=dpc)
  area  <- area1+runif(1)*area2
  taunovo[j] <- qnorm(area,mean=tau[j],sd=dpc)
}

p1<-rep(0,K-2)
p2<-rep(0,K-2)
p3<-rep(0,K-2)
p4<-rep(0,K-2)

for(j in 1:(K-2))
{
  p1[j] <- pnorm((tau[j+2] -tau[j+1])/dpc)
}
for(j in 1:(K-2))
{
  p2[j] <- pnorm((taunovo[j] -tau[j+1])/dpc)
}
for(j in 1:(K-2))
{
  p3[j] <- pnorm((taunovo[j+2] -taunovo[j+1])/dpc)
}
for(j in 1:(K-2))
{
  p4[j] <- pnorm((tau[j] -taunovo[j+1])/dpc)
}

##### Amostra lambda (GS elemento a elemento) ###

for (j in 1:n)
{

```

```

lambda[j] <- rgamma(1, (nu+i)/2, (nu+(L[j]-W[j,]*%*%theta)^2)/2)
)
R <- diag(lambda)

for(k in 1:n)
{
m <- W%*%theta
dp <- rep(1,n)
dp <- sqrt(ve/lambda)

if(y[k]==1)
{
ftau[k] <- pnorm((tau[1] - m[k])/dp[k])
ftaunovo[k] <- pnorm((taunovo[1] - m[k])/dp[k])
}
if(y[k]==2)
{
ftau[k] <- pnorm((tau[2] - m[k])/dp[k]) -
pnorm((tau[1] - m[k])/dp[k])
ftaunovo[k] <- pnorm((taunovo[2] - m[k])/dp[k]) -
pnorm((taunovo[1] - m[k])/dp[k])
}
if(y[k]==3)
{
ftau[k] <- pnorm((tau[3] - m[k])/dp[k]) -
pnorm((tau[2] - m[k])/dp[k])
ftaunovo[k] <- pnorm((taunovo[3] - m[k])/dp[k]) -
pnorm((taunovo[2] - m[k])/dp[k])
}
if(y[k]==4)
{
ftau[k] <- pnorm((tau[4] - m[k])/dp[k]) -
pnorm((tau[3] - m[k])/dp[k])
ftaunovo[k] <- pnorm((taunovo[4] - m[k])/dp[k]) -
pnorm((taunovo[3] - m[k])/dp[k])
}
if(y[k]==5)
{
ftau[k] <- 1 - pnorm((tau[4] - m[k])/dp[k])
ftaunovo[k] <- 1 - pnorm((taunovo[4] - m[k])/dp[k])
}
}

alfa <- exp(sum(log((p1-p2))-log((p3-p4)))+
sum(log(ftaunovo)-log(ftau)))
teste <- runif(1)
D <- min(1,alfa)

if(teste < D)
{
tau <- taunovo
##### Amostra nu (MH) #####

nuc <- 2
while(nuc < 3)

```

```

{
nuc <- rpois(1,nu)
}
pnuc <- n*{(nu/2)*log(nu/2)-lgamma(nu/2)+sum(((nu/2)-1)*log(lambda)+
(-nu/2)*lambda)} - 2*log(1+nu)
pnuc <- n*{(nuc/2)*log(nuc/2)-lgamma(nuc/2)+sum(((nuc/2)-1)*log(lambda)+
(-nuc/2)*lambda)} - 2*log(1+nuc)
pnu <- exp(pnuc)
pnuc <- exp(pnuc)
dnu <- dpois(nu,nu)
dnuc <- dpois(nuc,nu)
num <- pnuc*dnu
denom <- pnu*dnuc

ifelse((num > denom), (alfa <- 1), (alfa <- as.real(num/denom)))
teste <- runif(1)
if(teste < alfa)
{
nu <- nuc
contnu <- contnu+1
}

##### condicional para L ####

for(k in 1:n)
{
if(y[k]==1)
{
L[k] <-qnorm(pnorm(tau[1],m[k],sd=dp[k])*runif(1),mean=m[k],sd=dp[k])
vero[k] <-pnorm((tau[1] - m[k])/dp[k])
}
if(y[k]==2)
{
L[k] <-qnorm(pnorm(tau[1],m[k],sd=dp[k])+(pnorm(tau[2],m[k],sd=dp[k])
-pnorm(tau[1],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
vero[k] <-pnorm((tau[2] - m[k])/dp[k]) - pnorm((tau[1] - m[k])/dp[k])
}
if(y[k]==3)
{
L[k] <-qnorm(pnorm(tau[2],m[k],sd=dp[k])+(pnorm(tau[3],m[k],sd=dp[k])
-pnorm(tau[2],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
vero[k] <-pnorm((tau[3] - m[k])/dp[k]) - pnorm((tau[2] - m[k])/dp[k])
}
if(y[k]==4)
{
L[k] <- qnorm(pnorm(tau[3],m[k],sd=dp[k])+(pnorm(tau[4],m[k],sd=dp[k])
-pnorm(tau[3],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
vero[k] <-pnorm((tau[4] - m[k])/dp[k]) - pnorm((tau[3] - m[k])/dp[k])
}
if(y[k]==5)
{
L[k] <- qnorm(pnorm(tau[4],m[k],sd=dp[k]) +
(1-pnorm(tau[4],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
vero[k] <- (1 - pnorm(- m[k]/dp[k]))
}
}
}

```

```

}

verossim <- prod(vero)

##### atualização do desvio padrão da candidata #####
dc[i] <- c(tau[3])
ifelse(i > 20, dpc <- sd(dc[(i-20):i]), dpc <- 0.25)

##### Gera a saída no arquivo cadeia.txt #####

saida <- cbind(t(theta),t(tau),vu,nu,dpc,verossim)
colunas <- length(saida)

if(i > burn && (i-burn)%%jump == 0)
{
  write(saida, file="cadeia.txt", ncol=colunas, append=TRUE)
}
}

```

Programa 5. Rotina R para implementação do algoritmo NC no caso em que a variável latente tem distribuição normal.

```
#####
## Algoritmo NC #
## Variável latente com distribuição normal #
#####

dados <- read.table("dados.txt",h=T)
attach(dados)

##### Definição do tamanho da cadeia de saída #

amostra <- 5000
burn <- 5000
jump <- 50
iter <- jump*amostra+burn

#####

library(VGAM)
library(MASS)
library(mvtnorm)

y <- cor
n <- length(y)

### reorganização dos dados em 5 categorias ##

for(i in 1:n)
{
  if (y[i]<=5) y[i]<-1
}
for(i in 1:n)
{
  if (y[i] == 6) y[i]<- 2
}
for(i in 1:n)
{
  if (y[i] == 7) y[i]<- 3
}
for(i in 1:n)
{
  if (y[i] == 8) y[i]<- 4
}
for(i in 1:n)
{
  if (y[i] == 9) y[i]<- 5
}

## Matrizes de incidência de tratamentos e blocos #

trat <- c(rep(1,36),rep(2,36),rep(3,36))
bloco <- dados$bloco
XT <-matrix(0,n,3)
for(i in 1:n)
```

```

{
  XT[i,trat[i]]<-1
}
XB <-matrix(0,n,35)
for(i in 1:n)
{
  ifelse(bloco[i]==36, XB[i,] <- -1, XB[i,bloco[i]] <- 1)
}
W <- cbind(XT,XB)

##### objetos para o cálculo da verossimilhança #####

vero <- rep(1,n)
verossim <- NULL

##### objetos para a atualização dos par. de limiar #

nk <- tapply(L,y,length)
ftau <- rep(1,n-nk[1]-nk[5])
ftaunovo <- ftau

##### Valores iniciais arbitrários #####
vu <- 3
pnovo <- c(0.2,0.4,0.4)
p <- pnovo
taunovo <- c(0, 0.2, 0.6, 1)
tau <- taunovo
delta2 <- 0.5
delta <- sqrt(delta2)
L <- c(1:n)
for(i in 1:n)
{
  L[i]<- qnorm(pnorm(0)+pnorm(0)*runif(1))
}

##### prioris #####
se <- 1000
dre <- 20
dse <- 5
ru <- 3
su <- 5

##### Amostrador de Gibbs ###

for (i in 1:iter)
{

##### condicional para theta #####

V <- rbind(cbind(se*diag(3) , matrix(0,3,35)),
          cbind(matrix(0,35,3), vu*A))
S <- delta2*ginv(V)
theta <- ginv(t(W)%*%W + S)%*%t(W)%*%L
var <- delta2*ginv(t(W)%*%W+ S)
thetaest <- rmvnorm(1,mean = theta,sigma = var, method = "chol")

```



```

theta   <- t(thetaest)
beta    <- theta[1:3]
u       <- theta[4:38]

### condicional para variancia de blocos ###

c1 <- (35+2*ru)/2
c2 <- (t(u)%*%ginv(A)%*%u + 2*su)/2
vu <- rgamma(1,c1,c2)
vu <- as.real(1/vu)

# par. de aceleramento da convergência (reparametrização) ###

delta2 <- 1/rgamma(1,(n+38+5+2*dre)/2,(t(L-W*%theta)%*%(L-W*%theta)
+ 2*dse +t(theta)%*%ginv(V)%*%theta)/2)
delta  <- sqrt(delta2)

##### Amostragem dos par. de limiar #####

nkp <- 0.8*nk[2:(5-1)]
pnovo <- c(rdiric(1, shape = nkp ))
for(j in 2:(5-1))
{
  taunovo[j]<-sum(pnovo[1:(j-1)])
}
lfp <- nkp*log(p)
lfpnovo <- nkp*log(pnovo)

m <- W*%theta
for(k in 1:n)
{
  if(y[k]==1)
  {
    ftau[k] <- 1
    ftaunovo[k] <- 1
  }
  if(y[k]==2)
  {
    ftau[k] <- pnorm((tau[2] - m[k])/delta) -
      pnorm((tau[1] - m[k])/delta)
    ftaunovo[k] <- pnorm((taunovo[2] - m[k])/delta) -
      pnorm((taunovo[1] - m[k])/delta)
  }
  if(y[k]==3)
  {
    ftau[k] <- pnorm((tau[3] - m[k])/delta) -
      pnorm((tau[2] - m[k])/delta)
    ftaunovo[k] <- pnorm((taunovo[3] - m[k])/delta) -
      pnorm((taunovo[2] - m[k])/delta)
  }
  if(y[k]==4)
  {
    ftau[k] <- pnorm((tau[4] - m[k])/delta) -
      pnorm((tau[3] - m[k])/delta)
    ftaunovo[k] <- pnorm((taunovo[4] - m[k])/delta) -

```

```

                                pnorm((taunovo[3] - m[k])/delta)
        }
        if(y[k]==5)
        {
            ftau[k]      <- 1
            ftaunovo[k] <- 1
        }
    }

    lftau      <- log(ftau)
    lftaunovo  <- log(ftaunovo)
    alfa       <- exp(sum(lftaunovo-iftau)-sum(lfpnovo-lfp))

    teste <- runif(1)
    R      <- min(1,alfa)
    if(teste < R)
    {
        cont <- cont + 1
        p    <- pnovo
        tau  <- taunovo

##### condicional para L ####

    for(k in 1:n)
    {
        if(y[k]==1)
        {
            L[k] <-qnorm(pnorm(tau[1],m[k],sd=delta)+runif(1),mean=m[k],sd=delta)
            vero[k]<-pnorm((tau[1] - m[k])/delta)
        }
        if(y[k]==2)
        {
            L[k] <-qnorm(pnorm(tau[1],m[k],sd=delta)+(pnorm(tau[2],m[k],sd=delta)
            -pnorm(tau[1],m[k],sd=delta))*runif(1),mean=m[k],sd=delta)
            vero[k]<-pnorm((tau[2] - m[k])/delta) - pnorm((tau[1] - m[k])/delta)
        }
        if(y[k]==3)
        {
            L[k] <-qnorm(pnorm(tau[2],m[k],sd=delta)+(pnorm(tau[3],m[k],sd=delta)
            -pnorm(tau[2],m[k],sd=delta))*runif(1),mean=m[k],sd=delta)
            vero[k]<-pnorm((tau[3] - m[k])/delta) - pnorm((tau[2] - m[k])/delta)
        }
        if(y[k]==4)
        {
            L[k] <-qnorm(pnorm(tau[3],m[k],sd=delta)+(pnorm(tau[4],m[k],sd=delta)
            -pnorm(tau[3],m[k],sd=delta))*runif(1),mean=m[k],sd=delta)
            vero[k]<-pnorm((tau[4] - m[k])/delta) - pnorm((tau[3] - m[k])/delta)
        }
        if(y[k]==5)
        {
            L[k] <-qnorm(pnorm(tau[4],m[k],sd=delta)+
            (1 - pnorm(tau[4],m[k],sd=delta))*runif(1),mean=m[k],sd=delta)
            vero[k]<-(1 - pnorm(- m[k]/delta))
        }
    }
}

```

```
##### Cálculo da verossimilhança #####  
  
verossim <- prod(vero)  
  
##### Gera a saída no arquivo cadeia.txt #####  
  
saida <- cbind(t(theta),t(tau),delta2,vu,verossim)  
colunas <- length(saida)  
  
if(i > burn && (i-burn)%%jump == 0)  
{  
  write(saida, file="cadeia.txt", ncol=colunas, append=TRUE)  
}  
}
```

Programa 6. Rotina R para implementação do algoritmo NC no caso em que a variável latente tem distribuição t de Student.

```
#####
## Algoritmo NC #
## Variável latente com distribuição t #
#####

dados <- read.table("dados.txt",h=T)
attach(dados)

##### Definição do tamanho da cadeia de saída #

amostra <- 5000
burn <- 5000
jump <- 50
iter <- jump*amostra+burn

#####

library(VGAM)
library(MASS)
library(mvtnorm)

y <- cor
n <- length(y)

### reorganização dos dados em 5 categorias ##

for(i in 1:n)
{
  if (y[i]<=5) y[i]<-1
}
for(i in 1:n)
{
  if (y[i] == 6) y[i]<- 2
}
for(i in 1:n)
{
  if (y[i] == 7) y[i]<- 3
}
for(i in 1:n)
{
  if (y[i] == 8) y[i]<- 4
}
for(i in 1:n)
{
  if (y[i] == 9) y[i]<- 5
}

## Matrizes de incidência de tratamentos e blocos #

trat <- c(rep(1,36),rep(2,36),rep(3,36))
bloco <- dados$bloco
XT <-matrix(0,n,3)
for(i in 1:n)
```

```

{
  XT[i,trat[i]]<-1
}
XB <-matrix(0,n,35)
for(i in 1:n)
{
  ifelse(bloco[i]==36, XB[i,] <- -1, XB[i,bloco[i]] <- 1)
}
W <- cbind(XT,XB)

#### objetos para o cálculo da verossimilhança ####

vero      <- rep(1,n)
verossim  <- NULL

#### objetos para a atualização dos par. de limiar #

nk        <- tapply(L,y,length)
ftau      <- rep(1,n-nk[1]-nk[5])
ftaunovo  <- ftau

##### Valores iniciais arbitrários #####
vu        <- 3
pnovo     <- c(0.2,0.4,0.4)
p         <- pnovo
taunovo   <- c(0, 0.2, 0.6, 1)
tau       <- taunovo
delta2    <- 0.5
delta     <- sqrt(delta2)
nu        <- 8
pnu       <- 0.5
pnuc      <- 0.5
nuc       <- rpois(1,5)
lambda    <- c(rep(1,n))
R         <- diag(n)
L         <- c(1:n)
  for(i in 1:n)
  {
    L[i]<- qnorm(pnorm(0)+pnorm(0)*runif(1))
  }

##### prioris #####
se        <- 1000
dre       <- 20
dse       <- 5
ru        <- 3
su        <- 5

##### Amostrador de Gibbs ###

for (i in 1:iter)
{

##### condicional para theta ####

```

```

V      <- rbind(cbind(se+diag(3)      , matrix(0,3,35)),
               cbind(matrix(0,35,3), vu*A))
S      <- delta2*ginv(V)
theta  <- ginv(t(W)%*%R%*%W + S)%*%t(W)%*%R%*%L
var    <- delta2*ginv(t(W)%*%R%*%W+ S)
thetaest <- rmvnorm(1,mean = theta,sigma = var, method = "chol")
theta  <- t(thetaest)
beta   <- theta[1:3]
u      <- theta[4:38]

### condicional para variancia de blocos ###

c1 <- (35+2*ru)/2
c2 <- (t(u)%*%ginv(A)%*%u + 2*su)/2
vu <- rgamma(1,c1,c2)
vu <- as.real(1/vu)

## par. de aceleramento da convergência (reparametrização) ##

delta2 <- 1/rgamma(1, (n+38+5+2*dre)/2, (t(L-W%*%theta)%*%(L-W%*%theta) +
2*dse +t(theta)%*%ginv(V)%*%theta)/2)
delta  <- sqrt(delta2)

##### Amostra lambda (GS elemento a elemento) #####

lambda <- rgamma(n, (nu + 1)/2, (nu+(L-W%*%theta)^2)/2)
R      <- diag(lambda)

##### Amostra nu (MH) #####

nuc <- 2
while(nuc < 3)
{
  nuc <- rpois(1,nu)
}
pnu <- n*{(nu/2)*log(nu/2)-lgamma(nu/2)}+sum(((nu/2)-1)*log(lambda)+
(-nu/2)*lambda) - 2*log(1+nu)
pnuc <- n*{(nuc/2)*log(nuc/2)-lgamma(nuc/2)}+sum(((nuc/2)-1)*log(lambda)+
(-nuc/2)*lambda) - 2*log(1+nuc)
pnu <- exp(pnu)
pnuc <- exp(pnuc)
dnu <- dpois(nu,nu)
dnuc <- dpois(nuc,nu)
num <- pnu*dnu
denom <- pnu*dnuc

ifelse((num > denom),(alfa <- 1),(alfa <- as.real(num/denom)))
:
teste <- runif(1)
if(teste < alfa)
{
  nu <- nuc
}

##### Amostragem dos par. de limiar #####

```

```

nkp <- 0.8*nk[2:4]
pnovo <- c(rdiric(1, shape = nkp))

for(j in 2:4)
{
  taunovo[j]<-sum(pnovo[1:(j-1)])
}
lfp <- nkp*log(p)
lfpnovo <- nkp*log(pnovo)

m <- W%*%theta
for(k in 1:n)
{
  dp <- rep(1,n)
  dp <- sqrt(delta2/lambda)
  if(y[k]==1)
  {
    ftau[k] <- 1
    ftaunovo[k]<- 1
  }
  if(y[k]==2)
  {
    ftau[k] <- pnorm((tau[2] - m[k])/dp[k]) -
      pnorm((tau[1] - m[k])/dp[k])
    ftaunovo[k]<- pnorm((taunovo[2] - m[k])/dp[k]) -
      pnorm((taunovo[1] - m[k])/dp[k])
  }
  if(y[k]==3)
  {
    ftau[k] <- pnorm((tau[3] - m[k])/dp[k]) -
      pnorm((tau[2] - m[k])/dp[k])
    ftaunovo[k]<- pnorm((taunovo[3] - m[k])/dp[k]) -
      pnorm((taunovo[2] - m[k])/dp[k])
  }
  if(y[k]==4)
  {
    ftau[k] <- pnorm((tau[4] - m[k])/dp[k]) -
      pnorm((tau[3] - m[k])/dp[k])
    ftaunovo[k]<- pnorm((taunovo[4] - m[k])/dp[k]) -
      pnorm((taunovo[3] - m[k])/dp[k])
  }
  if(y[k]==5)
  {
    ftau[k] <- 1
    ftaunovo[k]<- 1
  }
}

lftau <- log(ftau)
lftaunovo<- log(ftaunovo)
alfa <- exp(sum(lftaunovo-lftau)-sum(lfpnovo-lfp))

teste <- runif(1)
D <- min(1,alfa)
if(teste < D)
{

```

```

p <- p novo
tau <- taunovo

##### condicional para L #####

m <- W%*%theta
for(k in 1:n)
{
  if (y[k]==1)
  {
    L[k] <-qnorm(pnorm(tau[1],m[k],sd=dp[k])*runif(1),mean=m[k],sd=dp[k])
    vero[k]<-pnorm((tau[1] - m[k])/dp[k])
  }
  if (y[k]==2)
  {
    L[k] <-qnorm(pnorm(tau[1],m[k],sd=dp[k])+(pnorm(tau[2],m[k],sd=dp[k])
    -pnorm(tau[1],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
    vero[k]<-pnorm((tau[2] - m[k])/dp[k]) - pnorm((tau[1] - m[k])/dp[k])
  }
  if (y[k]==3)
  {
    L[k] <-qnorm(pnorm(tau[2],m[k],sd=dp[k])+(pnorm(tau[3],m[k],sd=dp[k])
    -pnorm(tau[2],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
    vero[k]<-pnorm((tau[3] - m[k])/dp[k]) - pnorm((tau[2] - m[k])/dp[k])
  }
  if (y[k]==4)
  {
    L[k] <-qnorm(pnorm(tau[3],m[k],sd=dp[k])+(pnorm(tau[4],m[k],sd=dp[k])
    -pnorm(tau[3],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
    vero[k] <-pnorm((tau[4] - m[k])/dp[k]) - pnorm((tau[3] - m[k])/dp[k])
  }
  if (y[k]==5)
  {
    L[k] <-qnorm(pnorm(tau[4],m[k],sd=dp[k]) +
    (1-pnorm(tau[4],m[k],sd=dp[k]))*runif(1),mean=m[k],sd=dp[k])
    vero[k]<-(1 - pnorm(- m[k]/dp[k]))
  }
}
}

##### Cálculo da verossimilhança #####

verossim <- prod(vero)

##### Gera a saída no arquivo cadeia.txt #####

saida <- cbind(t(theta),t(tau),delta2,nu,vu,verossim)
colunas <- length(saida)

if(i > burn && (i-burn)%%jump == 0)
{
  write(saida, file="cadeia.txt", ncol=colunas, append=TRUE)
}
}

```