ESTIMAÇÃO PONTUAL EM REGRESSÃO BETA: ASPECTOS COMPUTACIONAIS

NÁTALY ADRIANA JIMÉNEZ MONROY

Orientador: Prof. Dr. Francisco Cribari-Neto
Co-orientador: Prof. Dr. Klaus Leite Pinto Vasconcellos

Área de Concentração: Estatística Aplicada

Dissertação submetida como requerimento parcial para obtenção do grau de Mestre em Estatística pela Universidade Federal de Pernambuco

Recife, fevereiro de 2007

Monroy, Nátaly Adriana Jiménez.
Estimação pontual em regressão Beta: aspectos computacionais / Nátaly Adriana Jiménez Monroy – Recife : O autor, 2007.
xii, 95 folhas. : il., fig., tab.

Dissertação (mestrado) – Universidade Federal de Pernambuco. CCEN. Estatística, 2007.

Inclui bibliografia.

1. Análise de regressão. 2. Otimização 3. Máxima verossimilhança. I. Título.

519.536 CDD (22.ed.) MEI2007-009

Universidade Federal de Pernambuco
Pós-Graduação em Estatística

15 de fevereiro de 2007
(data)

Nós recomendamos que a dissertação de mestrado de autoria de

Nátaly Jiménez Monroy

intitulada

"Estimação Pontual em Regressão Beta: Aspectos Computacionais"

seja aceita como cumprimento parcial dos requerimentos para o grau de Mestre em Estatística.

Coordenador da Pós-Graduação em Estatística

Banca Examinadora:

Francisco Cribari Neto orientador

Silvia Lopes de Paula Ferrari (USP)

Renato José de Sobral Cintra

Este documento será anexado à versão final da dissertação.

Typeset by LaTeX

A Deus e a Maria Santíssima.

À minha adorada família.

AGRADECIMENTOS

A Deus, por me dar a vida, a família e as ótimas oportunidades das quais tenho desfrutado.

Ao meu amor, Fabio Fajardo Molinares, por todos os momentos de felicidade que me tem proporcionado e pelo apoio e incentivo constantes.

Aos meus pais, Salvador e Teresa, e às minhas irmãs Gigi e Teisy, por sua constante voz de ânimo e porque, mesmo estando longe, seu amor e sua força me acompanham aonde quer que eu vá.

Aos meus tios, especialmente a Esperanza, Guillermo, Luis, Rosita, Aurorita e Margarita; e às minhas primas Patricia, Diana, Marcela, Karina e Zulma, pelas boas energias que sempre me enviam.

Aos professores Francisco Cribari-Neto e Klaus Leite Pinto Vasconcellos pela orientação, sugestões e valiosas recomendações que tornaram possível este trabalho.

Aos professores do Mestrado em Estatística da UFPE, especialmente Klaus Vasconcellos pelo magnífico curso de inferência estatística, Francisco Cribari-Neto pelo interessante curso de Estatística Aplicada e Renato J. Cintra pelo exemplar curso de Probabilidade.

Ao professor Luis Alberto López, do Departamento de Estatística da Universidad Nacional de Colombia, pelo contínuo apoio e pela confiança depositada em mim.

Aos meus amigos na Colômbia, especialmente a Luz Dary, Wilson, Pacho, Camilo, Doris,

David, Edgar, Angela, Andrea e Iván D., pelo apoio e pela constante preocupação com meu bem-estar.

Aos meus amigos da colônia colombiana no Brasil, especialmente a Edwin, Patricia, Carolina, Diana, Alexandra, Ricardo e José Barba pela solidariedade, amizade e diversão.

A Solange, Marcelo, Gilvan, Márcia, Alexandre e Andrea, pela amizade e pelos inesquecíveis momentos de diversão que tornaram mais amena minha permanência em Recife.

Aos colegas do mestrado, pelas horas de trabalho acadêmico compartilhado.

A Valéria Bittencourt, pela presteza e carinho com que sempre me ofereceu sua ajuda.

À CAPES, pelo apoio financeiro.

RESUMO

A classe de modelos de regressão beta é de grande utilidade em situações de modelagem onde o objetivo reside no estudo da relação entre uma variável de interesse que assume continuamente valores no intervalo $(0, 1)$ e outras variáveis que afetam seu comportamento através de uma estrutura de regressão. A presente dissertação dedica-se a estudar aspectos computacionais inerentes à estimação pontual dos parâmetros do modelo de regressão beta proposto por Ferrari & Cribari-Neto (2004) através da avaliação de diferentes métodos de otimização não-linear que podem ser utilizados para maximizar numericamente a função de log-verossimilhança.

Nós mostramos, através de simulações de Monte Carlo e de estimações com conjuntos de dados reais, que os métodos de otimização não-linear que usam informação relativa à matriz hessiana, como é o caso dos métodos de Newton e BFGS, são os mais eficientes no que tange à maximização da função de log-verossimilhança do modelo de regressão beta. Isso ocorre devido à sua rapidez, precisão e robustez frente a perturbações comumente verificadas em situações práticas, tais como presença de pontos de alavanca e elevada correlação entre variáveis regressoras.

Palavras-chave: Regressão beta, Otimização, Máxima verossimilhança.

ABSTRACT

We consider the beta regression model proposed by Ferrari & Cribari-Neto (2004), which is tailored to situations where the response is restricted to the standard unit interval and has a regression structure involving regressors and unknown parameters. Our chief interest is the evaluation of several nonlinear optimization methods in the context of maximizing the beta regression log-likelihood function. The numerical evidence from Monte Carlo simulations and empirical analyses based on real data favors the Newton and BFGS algorithms, which are fast, accurate and behave well even in unfavorable situations such as the existence of leverage points and high correlation amongst regressors.

Key words: Beta regression, Optimization, Maximum likelihood.

Índice

Lista de Figuras

Lista de Tabelas

CAPÍTULO 1

Introdução

A classe de modelos de regressão beta é de grande utilidade em situações de modelagem onde o objetivo reside no estudo da relação entre uma variável de interesse que assume continuamente valores no intervalo $(0, 1)$ e outras variáveis que afetam seu comportamento através de uma estrutura de regressão. Na literatura estatística, é possível encontrar diferentes especificações desse modelo; ver, e.g., Paolino (2001), Kieschnick & McCullough (2003), Vasconcellos & Cribari-Neto (2005) e Smithson & Verkuilen (2006). A classe de modelos estudada nesta dissertação é aquela especificada por Ferrari & Cribari-Neto (2004), a qual é indexada por parâmetros de locação e dispersão. A resposta média é relacionada a um preditor linear que incorpora as covariáveis e parâmetros desconhecidos através de uma função de ligação.

Um aspecto importante na estimação dos parâmetros por máxima verossimilhança é a escolha de um processo numérico eficaz de maximização da função de log-verossimilhança, dado que podem surgir problemas numéricos dependendo da forma da função. Por exemplo, se a função for relativamente plana na região próxima ao ponto de máximo, o algoritmo pode ser interrompido longe do verdadeiro valor do parâmetro. Adicionalmente, o valor máximo obtido pode depender do ponto escolhido para iniciar o processo iterativo; se o ponto inicial estiver muito longe do ponto maximizador, pode ser difícil localizar o valor máximo da função.

O objetivo da presente dissertação é estudar aspectos computacionais inerentes à estimação pontual dos parâmetros que definem a classe de modelos de regressão beta proposta por Ferrari & Cribari-Neto (2004). Nós avaliamos os desempenhos de alguns algoritmos de otimização não-linear sob diferentes condições tanto via simulação de Monte Carlo quanto através de aplicações que usam dados reais. A estrutura do restante da dissertação encontra-se descrita a seguir. No Capítulo 2 é apresentado o modelo de regressão beta, assim como aspectos da estimação pontual de seus parâmetros por máxima verossimilhança e algumas de suas propriedades assintóticas. No terceiro capítulo são apresentados alguns métodos de otimização de grande utilidade para a maximização da função de log-verossimilhança do modelo. O quarto capítulo contém os resultados das simulações de Monte Carlo realizadas para a avaliação dos métodos de otimização usados na estimação pontual do modelo. No Capítulo 5 são avaliados os desempenhos dos métodos considerados nos Capítulos 3 e 4 a partir de aplicações com dados reais. Finalmente, as conclusões do presente trabalho encontram-se agrupadas no Capítulo 6.

CAPÍTULO 2

Regressão Beta

2.1 Introdução

Em diferentes áreas de aplicação da Estatística, é prática comum analisar a influência de diversas variáveis sobre a esperança condicional de porcentagens, proporções ou razões. Kieschnick & McCullough (2003) listaram vários estudos que incluem tais variáveis e destacaram a falta de consenso entre os pesquisadores no que diz respeito aos modelos distribucionais e aos modelos de regressão que devem ser usados na modelagem. Segundo estes autores, alguns dos modelos empregados são, inclusive, de utilidade duvidosa. Por exemplo, freqüentemente são ajustados modelos de regressão normal linear, os parâmetros que indexam o modelo sendo estimados pelo método de mínimos quadrados ordinários (MQO). Esta aproximação não é adequada, pois, entre outras razões, a variância da resposta deve ser

variável, aproximando-se de zero quando a média se aproxima de qualquer um dos pontos extremos do intervalo $(0, 1)$.

Em resposta a este problema surgiu o modelo de regressão beta, onde a resposta, que é continuamente distribuída no intervalo $(0, 1)$, segue lei beta e depende de outras variáveis através de uma estrutura de regressão. Mais geralmente, se a variável dependente estiver restrita ao intervalo (a, b), em que $a < b$ são escalares conhecidos, pode-se modelar a variável transformada $(y-a)/(b-a)$. É plausível assumir que a resposta segue distribuição na família beta, caracterizada por sua flexibilidade para produzir distribuições unimodais, uniformes ou bimodais simétricas e assimétricas; adicionalmente, esta família é uma das mais usadas em modelagens paramétricas (Johnson, Kotz & Balakrishnan 1995).

Segundo Smithson & Verkuilen (2006), os três primeiros exemplos de regressão beta vieram da literatura na área de Economia, onde Brehm & Gates (1993) modelaram proporções usando a parametrização padrão das distribuições beta, que complexifica a formulação do modelo de regressão e dificulta a interpretação dos parâmetros. Posteriormente, Paolino (2001) mostrou, por simulação de Monte Carlo, que a aproximação do ajuste de modelos lineares por mínimos quadrados pode levar a inferências erradas sobre os efeitos das covariáveis. O autor propôs estimar o efeito da variável explicativa sobre a média e sobre a dispersão da variável resposta especificando os parâmetros da distribuição da resposta em termos da média e da dispersão, tornando mais simples a interpretação dos parâmetros e a inferência sobre os efeitos das covariáveis sobre a resposta esperada e sobre a dispersão. Buckley (2003) apresentou uma alternativa bayesiana ao modelo proposto por Paolino (2001) e usou amostragem MCMC (Markov Chain Monte Carlo) para extrair amostras

de uma distribuição beta *a posteriori*.

Recentemente, Ferrari & Cribari-Neto (2004) propuseram um modelo de regressão beta usando uma parametrização da distribuição beta indexada pelos parâmetros de locação e dispersão. Estes parâmetros podem ser interpretados em termos da média das observações, que é modelada usando um preditor linear que relaciona a resposta média a covariáveis e parâmetros desconhecidos através de uma função de ligação, como acontece nos modelos lineares generalizados. O comportamento das estimativas dos parâmetros deste modelo em amostras pequenas foi estudado por Ospina, Cribari-Neto & Vasconcellos (2006) através da avaliação de diferentes estratégias de estimação pontual e intervalar.

Além de apresentar detalhes da estimação dos parâmetros por máxima verossimilhança, Ferrari & Cribari-Neto (2004) desenvolveram um conjunto compreensivo de ferramentas de inferência, tais como testes de hipóteses e medidas de diagnóstico.

2.2 O Modelo

No modelo de regressão beta, assume-se que a variável resposta segue distribuição beta com densidade

$$f(y;p,q) = \frac{\Gamma(p+q)}{\Gamma(p)\Gamma(q)} y^{p-1}(1-y)^{q-1}, \quad y \in (0,1), \tag{2.1}$$

onde $p > 0$, $q > 0$ e $\Gamma(\cdot)$ é a função gama. A média e a variância de y são, respectivamente,

$$E(y) = \frac{p}{p+q} \qquad \text{e} \qquad \text{var}(y) = \frac{pq}{(p+q)^2(p+q+1)}.$$

Sejam $y_1, \ldots, y_n$ variáveis aleatórias independentes, em que cada y_t, $t = 1, \ldots, n$, segue distribuição beta de média μ_t, com densidade como a apresentada em (2.1), e sejam $x_{t1}, \ldots, x_{tk}$

observações de k covariáveis fixas ($k < n$). Uma possível especificação do modelo assume que a média é uma função linear das variáveis explicativas, i.e.,

$$E(y_t) = \mu_t = x_t^\top \beta,$$

em que $x_t^\top = (x_{t1}, \ldots, x_{tk})$ e β é um vetor de parâmetros desconhecidos. Esta formulação não é muito adequada, pois não restringe os valores assumidos pela média condicional, exigindo a imposição de restrições sobre os valores das variáveis explicativas para fornecer resultados plausíveis (Kieschnick & McCullough 2003). Outra formulação, mais adequada, consiste em assumir que

$$E(y_t) = \mu_t = h(\eta_t) = \frac{1}{1 + \exp(-\eta_t)},$$

em que $\eta_t = x_t^\top \beta$. A função $h^{-1}(\cdot)$ dada aqui é um caso especial do que se conhece como *função de ligação*. É importante notar que esta especificação restringe a média condicional da variável de interesse ao intervalo $(0, 1)$, o que é apropriado no presente contexto.

Ferrari & Cribari-Neto (2004) definem a estrutura de regressão usando uma reparametrização da densidade apresentada em (2.1), indexando-a pelos parâmetros de locação e precisão μ e ϕ, respectivamente, sendo $\mu = \frac{p}{p+q}$ e $\phi = p + q$. Assim,

$$E(y) = \mu \qquad \text{e} \qquad \text{var}(y) = \frac{V(\mu)}{1 + \phi},$$

em que $V(\mu) = \mu(1 - \mu)$, de tal forma que μ é a média da variável resposta e ϕ pode ser interpretado como um *parâmetro de precisão*, no sentido de que, para μ fixo, quanto maior for o valor de ϕ, menor será a variância de y. A densidade da variável resposta, y, pode ser escrita como

$$f(y; \mu, \phi) = \frac{\Gamma(\phi)}{\Gamma(\mu\phi)\Gamma((1-\mu)\phi)} y^{\mu\phi - 1}(1 - y)^{(1-\mu)\phi - 1}, \quad 0 < y < 1, \tag{2.2}$$

onde $0 < \mu < 1$ e $\phi > 0$. Esta densidade pode apresentar muitas formas diferentes dependendo dos valores assumidos pelos parâmetros μ e ϕ. Em particular, ela pode ser simétrica (quando $\mu = 1/2$) ou assimétrica (quando $\mu \neq 1/2$).

O modelo é definido assumindo que y_t segue distribuição beta, com densidade (2.2), de média μ_t e precisão desconhecida ϕ e que

$$g(\mu_t) = \sum_{i=1}^{k} x_{ti}\beta_i = \eta_t, \tag{2.3}$$

em que $\beta = (\beta_1, \ldots, \beta_k)^\top$ é um vetor de parâmetros desconhecidos ($\beta \in I\!R^k$); as covariáveis $x_{t1}, \ldots, x_{tk}$ são assumidas fixas e conhecidas. Aqui, $g(\cdot)$ é uma função de ligação estritamente monótona e duas vezes diferenciável com domínio no intervalo $(0, 1)$ e contra-domínio $I\!R$. Cabe notar que a variância de y_t é função de μ_t e, como conseqüência, dos valores das covariáveis. Assim, variâncias não-constantes da resposta são naturalmente acomodadas no modelo.

No modelo de regressão beta podem ser usadas diferentes funções de ligação, tais como a função logit $g(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$, a função probit $g(\mu) = \Phi^{-1}(\mu)$, onde $\Phi(\cdot)$ representa a função de distribuição normal padrão e a função log-log complementar $g(\mu) = \log\{-\log\{1-\mu\}\}$. Para maiores detalhes sobre funções de ligação, ver McCullagh & Nelder (1989). Particularmente, quando usamos a função logit, a média pode ser escrita como

$$\mu_t = \frac{e^{x_t^\top \beta}}{1 + e^{x_t^\top \beta}}.$$

Aqui, os parâmetros do preditor linear têm interpretação importante. Suponha que o valor do i-ésimo regressor é incrementado em uma unidade e que todas as outras covariáveis permanecem inalteradas. Seja $\mu^\ddagger$ a média de y sob os novos valores das covariáveis, enquanto

μ denota a média de y sob os valores originais das covariáveis. Então,

$$e^{\beta_i} = \frac{\mu^{\ddagger}/(1-\mu^{\ddagger})}{\mu/(1-\mu)},$$

i.e., $\exp\{\beta_i\}$ é igual à razão de chances (Ferrari & Cribari-Neto 2004).

2.3 Estimação dos Parâmetros

A função de log-verossimilhança baseada em uma amostra de n observações independentes é dada por

$$\ell(\beta, \phi) = \sum_{t=1}^{n} \ell_t(\mu_t, \phi), \tag{2.4}$$

onde

$$\ell_t(\mu_t, \phi) = \log\Gamma(\phi) - \log\Gamma(\mu_t\phi) - \log\Gamma((1-\mu_t)\phi) + (\mu_t\phi - 1)\log y_t$$
$$+ \{(1-\mu_t)\phi - 1\}\log(1-y_t),$$

com μ_t definido segundo (2.3). Como (2.4) é uma reparametrização invertível da log-verossimilhança beta, pode-se garantir que o estimador de máxima verossimilhança é único (Ospina et al. 2006).

A função escore, obtida derivando-se a função de log-verossimilhança com respeito aos parâmetros desconhecidos, é dada por $(U_\beta(\beta,\phi)^\top, U_\phi(\beta,\phi))^\top$, onde

$$U_\beta(\beta,\phi) = \frac{\partial \ell_t(\beta,\phi)}{\partial \beta_i} = \sum_{t=1}^{n} \frac{\partial \ell_t(\mu_t,\phi)}{\partial \mu_t}\frac{d\mu_t}{d\eta_t}\frac{\partial \eta_t}{\partial \beta_i}.$$

Note que $\dfrac{d\mu_t}{d\eta_t} = \dfrac{dg^{-1}(\eta_t)}{d\eta_t} = \dfrac{1}{g'(\mu_t)}$. Também,

$$\frac{\partial \ell_t(\mu_t, \phi)}{\partial \mu_t} = \phi \left[\log \frac{y_t}{1-y_t} - \{\psi(\mu_t\phi) - \psi((1-\mu_t)\phi)\} \right],$$

onde $\psi(\cdot)$ é a função digama, ou seja, $\psi(z) = \frac{d \log \Gamma(z)}{dz}$, para $z > 0$. Sejam $y_t^* = \log\left(\frac{y_t}{1-y_t}\right)$ e $\mu_t^* = \psi(\mu_t\phi) - \psi((1-\mu_t)\phi)$. Então,

$$U_\beta(\beta, \phi) = \phi \sum_{t=1}^{n} (y_t^* - \mu_t^*) \frac{1}{g'(\mu_t)} x_{ti}$$

ou, em notação matricial,

$$U_\beta(\beta, \phi) = \phi X^\top T(y^* - \mu^*), \tag{2.5}$$

em que X é uma matriz $n \times k$ cuja t-ésima linha é $x_t^\top$, $y^* = (y_1^*, \ldots, y_n^*)^\top$, $\mu^* = (\mu_1^*, \ldots, \mu_n^*)^\top$ e $T = \text{diag}\{1/g'(\mu_1), \ldots, 1/g'(\mu_n)\}$. De forma similar, pode-se mostrar que a função escore para o parâmetro ϕ pode ser escrita como

$$U_\phi(\beta, \phi) = \sum_{t=1}^{n} \{\mu_t(y_t^* - \mu_t^*) + \log(1-y_t) - \psi((1-\mu_t)\phi) + \psi(\phi)\}. \tag{2.6}$$

Os estimadores de máxima verossimilhança (EsMV) de β e ϕ podem ser obtidos igualando (2.5) e (2.6) a zero e resolvendo o sistema de equações resultante. Dado que a solução destas equações não possui forma fechada, ela deve ser obtida pela maximização numérica da função de log-verossimilhança usando algoritmos de otimização não-linear. Estes algoritmos, que são tipicamente de natureza iterativa, requerem a especificação de um valor inicial. Ferrari & Cribari-Neto (2004) sugerem usar como valor inicial do vetor β a estimativa obtida por mínimos quadrados do vetor de coeficientes da regressão linear das respostas transformadas $g(y_1), \ldots, g(y_n)$ em X, ou seja, $(X^\top X)^{-1} X^\top z$, onde $z = (g(y_1), \ldots, g(y_n))$. No que tange ao parâmetro de dispersão, os autores sugerem usar como valor inicial

$$\frac{1}{n} \sum_{t=1}^{n} \frac{\check{\mu}_t(1-\check{\mu}_t)}{\check{\sigma}_t^2} - 1,$$

onde $\check{\mu}_t$ é obtido pela aplicação de $g^{-1}(\cdot)$ ao t-ésimo valor ajustado da regressão linear de z em X, isto é, $\check{\mu}_t = g^{-1}(x_t^\top (X^\top X)^{-1} X^\top z)$ e $\check{\sigma}_t^2 = \dfrac{\check{e}^\top \check{e}}{(n-k)\{g'(\check{\mu}_t)\}^2}$, sendo $\check{e} = z - (X^\top X)^{-1} X^\top z$ o vetor de resíduos de mínimos quadrados da regressão linear sob a resposta transformada.

Para determinar a variabilidade das estimativas dos parâmetros do modelo de regressão beta, Ferrari & Cribari-Neto (2004) obtiveram uma expressão para a matriz de informação de Fisher. Sejam $W = \text{diag}\{w_1, \ldots, w_n\}$, com

$$w_t = \phi\{\psi'(\mu_t\phi) + \psi'((1-\mu_t)\phi)\}\frac{1}{\{g'(\mu_t)\}^2},$$

$c = (c_1, \ldots, c_n)^\top$ com elemento típico $c_t = \phi\{\psi'(\mu_t\phi)\mu_t - \psi'((1-\mu_t)\phi)(1-\mu_t)\}$, em que $\psi'(\cdot)$ é a função trigama, e $D = \text{diag}\{d_1, \ldots, d_n\}$, onde $d_t = \psi'(\mu_t\phi)\mu_t^2 + \psi'((1-\mu_t)\phi)(1-\mu_t^2 - \psi'(\phi))$. A matriz de informação de Fisher pode ser escrita como

$$K = K(\beta, \phi) = \begin{pmatrix} K_{\beta\beta} & K_{\beta\phi} \\ K_{\phi\beta} & K_{\phi\phi} \end{pmatrix}, \tag{2.7}$$

onde $K_{\beta\beta} = \phi X^\top W X$, $K_{\beta\phi} = K_{\phi\beta}^\top = X^\top T c$ e $K_{\phi\phi} = \text{tr}(D)$, onde $\text{tr}(\cdot)$ denota o traço de uma matriz quadrada. Em contraste ao que ocorre na classe dos modelos lineares generalizados, os parâmetros β e ϕ não são ortogonais.

É possível obter aproximações para as quantidades w_t, c_t, d_t, $t = 1, \ldots, n$, que facilitam o cálculo da matriz de informação de Fisher quando $\mu_t\phi$ e $(1-\mu_t)\phi$ são grandes. Pode-se mostrar que, para $z \to \infty$,

$$\psi(z) = \log(z) - \frac{1}{2z} - \frac{1}{12z^2} + \frac{1}{120z^4} + \cdots,$$
$$\psi'(z) = \frac{1}{z} - \frac{1}{2z^2} + \frac{1}{6z^3} - \frac{1}{30z^5} + \cdots.$$

Desta forma, se $\mu_t\phi$ e $(1-\mu_t)\phi$ são suficientemente grandes, i.e., se ϕ é grande, então, ao utilizar as expansões para a função digama e sua primeira derivada, têm-se as aproximações

$$\begin{aligned} w_t &\approx \frac{1}{\phi\mu_t(1-\mu_t)} = \frac{1}{\phi V(\mu_t)}, \\ c_t &\approx \phi\left[\mu_t\frac{1}{\phi\mu_t(1-\mu_t)} - \frac{1}{\phi(1-\mu_t)}\right] = 0, \\ d_t &\approx \frac{(1-\mu)}{\phi} + \frac{\mu}{\phi} - \frac{1}{\phi} = 0. \end{aligned}$$

Logo, $K_{\beta\beta} \approx X^\top W_{\beta\beta} X$, com $W_{\beta\beta} = \text{diag}\left(\left(\frac{d\mu_t}{d\eta_t}\right)^2 \frac{\phi}{V(\mu_t)}\right)$, $K_{\beta\phi} = K_{\phi\beta}^\top \approx \mathbf{0}$ e $K_{\phi\phi} \approx 0$, onde $\mathbf{0}$ é um vetor de zeros de dimensão $k \times 1$. Desta forma, à medida que ϕ aumenta, os blocos $K_{\beta\phi}$ e $K_{\phi\beta}$ tendem a zero, assim como $K_{\phi\phi}$. Porém, quando $\phi \to \infty$, a derivada em (2.6) tende a zero e a curvatura tende para zero, i.e., a função de log-verossimilhança torna-se menos curva, o que faz com que ϕ seja de difícil estimação.

Usando a expressão padrão para a inversa de matrizes particionadas (ver Rao (1973) e Harville (1997), entre outros), é possível mostrar que a inversa da matriz de informação de Fisher (2.7) é

$$K^{-1} = K(\beta,\phi)^{-1} = \begin{pmatrix} K^{\beta\beta} & K^{\beta\phi} \\ K^{\phi\beta} & K^{\phi\phi} \end{pmatrix}, \tag{2.8}$$

com

$$K^{\beta\beta} = (X^\top W_{\beta\beta} X)^{-1}\left\{I_k + \frac{X^\top T c c^\top T^\top X (X^\top W_{\beta\beta} X)^{-1}}{\gamma}\right\} \tag{2.9}$$

$$= K_{\beta\beta}^{-1}\left\{I_k + \frac{K_{\beta\phi}K_{\phi\beta}K_{\beta\beta}^{-1}}{\gamma}\right\}, \tag{2.10}$$

em que $\gamma = \text{tr}(\text{diag}(d_t)) - c^\top T^\top X (X^\top W_{\beta\beta} X)^{-1} X^\top T c = \text{tr}(\text{diag}(d_t)) K_{\phi\beta} K_{\beta\beta}^{-1} K_{\beta\phi}$. Adicionalmente,

$$K^{\beta\phi} = (K^{\phi\beta})^\top = -\frac{1}{\gamma}(X^\top W_{\beta\beta} X)^{-1} X^\top T c = -\frac{1}{\gamma} K_{\beta\beta}^{-1} K_{\beta\phi}$$

e $K^{\phi\phi} = \frac{1}{\gamma}$. Aqui, I_k representa a matriz identidade de dimensão k. Sob condições de regularidade, quando o tamanho amostral é grande,

$$\begin{pmatrix} \widehat{\beta} \\ \widehat{\phi} \end{pmatrix} \overset{\mathcal{A}}{\sim} \mathcal{N}_{k+1} \left(\begin{pmatrix} \beta \\ \phi \end{pmatrix}, K(\beta, \phi)^{-1} \right),$$

com $\overset{\mathcal{A}}{\sim}$ denotando assintoticamente distribuído, $\widehat{\beta}$ e $\widehat{\phi}$ sendo os EsMV de β e ϕ, respectivamente, e $\mathcal{N}_{k+1}$ denotando a distribuição normal $(k+1)$-variada.

A partir da normalidade assintótica e dada a consistência do estimador de máxima verossimilhança, podem ser construídos intervalos de confiança assintóticos para os parâmetros do modelo de regressão beta. Assim, para $r = 1, \ldots, k$,

$$\left(\widehat{\beta}_r - z_{1-\alpha/2}(K(\widehat{\beta}, \widehat{\phi})^{rr})^{1/2}, \widehat{\beta}_r + z_{1-\alpha/2}(K(\widehat{\beta}, \widehat{\phi})^{rr})^{1/2} \right)$$

e

$$\left(\widehat{\phi} - z_{1-\alpha/2}(K(\widehat{\beta}, \widehat{\phi})^{\phi\phi})^{1/2}, \widehat{\phi} + z_{1-\alpha/2}(K(\widehat{\beta}, \widehat{\phi})^{\phi\phi})^{1/2} \right)$$

são intervalos de confiança assintóticos para β_r e ϕ, respectivamente, de cobertura $100(1-\alpha)\%$, $0 < \alpha < 1/2$. As variâncias assintóticas de $\widehat{\beta}_r$ e $\widehat{\phi}$ são $K(\widehat{\beta}, \widehat{\phi})^{rr}$ e $K(\widehat{\beta}, \widehat{\phi})^{\phi\phi}$, sendo $K(\widehat{\beta}, \widehat{\phi})^{rr}$ o elemento (r, r) da matriz $K^{\beta\beta}$ avaliado em $(\beta^\top, \phi)^\top$.

Usando aproximações obtidas a partir da normalidade assintótica do estimador de máxima verossimilhança, Ferrari & Cribari-Neto (2004) obtiveram algumas estatísticas úteis para a realização de testes de hipóteses. Adicionalmente, os autores apresentaram algumas medidas de diagnóstico para o modelo de regressão beta. Posteriormente, Espinheira, Ferrari & Cribari-Neto (2006) propuseram medidas de diagnóstico baseadas em resíduos derivados do algoritmo iterativo escore de Fisher na estimação do parâmetro β quando ϕ é fixo.

2.4 Implementação Computacional

O pacote `betareg` do software R (`http://www.r-project.org`), desenvolvido e mantido por Alexandre Simas, implementa a classe de modelos de regressão beta apresentada neste capítulo. Este pacote contém rotinas para estimação, testes de hipóteses e análise de diagnóstico. Estimação pontual é realizada usando o método quasi-Newton BFGS.

CAPÍTULO 3

Métodos de Otimização Numérica

3.1 Introdução

Métodos de otimização de funções são de grande importância na resolução de problemas estatísticos cujas soluções correspondem a pontos de máximo ou de mínimo de funções de interesse. Algumas técnicas de otimização precisam apenas da função objetivo, sendo chamadas de *métodos de procura direta*, como é o caso do método simplex; outros métodos requerem a avaliação explícita do vetor de derivadas parciais (vetor gradiente) da função a ser otimizada; estes são chamados *métodos gradiente*, dentre os quais pode-se citar o método de gradiente conjugado. Existem também técnicas que envolvem a estimação da matriz hessiana da função, sendo conhecidas como *métodos de métrica variável* ou *métodos quasi-Newton*.

O procedimento geral dos algoritmos de otimização consiste em, a partir de um ponto inicial, determinar a melhor direção e o tamanho do passo para "caminhar" pela superfície de uma função até o ponto ótimo, seja este um ponto de máximo ou de mínimo. O processo iterativo é finalizado quando algum critério de parada é satisfeito. A convergência de cada um destes métodos é afetada pelas freqüentes mudanças de direção necessárias, pela possibilidade de que a direção em uma dada iteração seja quase perpendicular à direção até o ponto ótimo e pelo tamanho do passo na direção do ponto ótimo, determinado pelo processo interno que ocorre durante a execução do algoritmo conhecido como *busca em linha*. Adicionalmente, alguns métodos podem ser ineficientes quando a aproximação de primeira ordem da função não é muito adequada; neste caso, uma aproximação de segunda ordem deve ser procurada (Khuri 1993).

Os métodos de otimização aqui apresentados são os do tipo gradiente, quasi-Newton e simplex. Embora todos estes sejam métodos de minimização de funções, eles também podem ser usados para encontrar o ponto de máximo apenas trocando o sinal da função e minimizando esta função modificada.

3.2 Métodos Gradiente

Há duas grandes famílias de algoritmos para minimização multidimensional que requerem o cálculo do gradiente (primeiras derivadas). A primeira delas é conhecida sob o nome de *métodos de gradiente conjugado* e requer armazenamento apenas de ordem d, além do cálculo de derivadas e de um procedimento de sub-minimização unidimensional, onde d é a dimensão do problema. A segunda família é conhecida pelo nome de métodos quasi-

Newton e requer armazenamento de ordem d^2, assim como um processo de sub-minimização unidimensional e o cálculo de derivadas de segunda ordem. A seguir serão descritos alguns dos algoritmos mais conhecidos pertencentes a cada família (Press, Teukolsky, Vetterling & Flannery 1992).

3.2.1 Método de Gradiente Conjugado

O primeiro método de gradiente conjugado não-linear foi introduzido por Fletcher & Reeves (1964) e é uma das primeiras técnicas conhecidas de resolução de problemas de otimização de grande escala. Muitas variantes deste método têm sido propostas. Entre as principais características destes algoritmos estão sua convergência quadrática e seu baixo custo computacional dado que cada iteração requer apenas a avaliação da função objetivo e de seu gradiente, não sendo realizadas operações matriciais (Nocedal & Wright 1999).

A convergência quadrática de um método iterativo implica que, para funções quadráticas, o ponto mínimo será localizado de forma exata em um número finito de iterações. Para funções gerais, quando as iterações se aproximam do mínimo, a aproximação quadrática da função se torna melhor e a convergência é garantida. Adicionalmente, mesmo em regiões afastadas do mínimo, o método consegue lidar com situações complexas, como a presença de grandes vales, levando em conta a curvatura da função.

Fletcher & Reeves (1964) desenvolveram uma fórmula recursiva simples que produz uma seqüência de direções mutuamente conjugadas. Um conjunto de vetores não-nulos $\{p_0, \ldots, p_l\}$ é dito ser conjugado com respeito à matriz G simétrica positiva-definida se

$$p_i^\top G p_j = 0 \quad \text{para todo } i \neq j.$$

Qualquer conjunto de vetores satisfazendo esta propriedade também é linearmente independente. Estas propriedades das direções são importantes uma vez que a minimização é realizada sucessivamente ao longo das direções individuais em um conjunto conjugado. Por exemplo, se a matriz G for diagonal, os contornos da função são elipses cujos eixos estão alinhados com as direções coordenadas. Assim, pode-se achar o ponto de mínimo realizando minimizações unidimensionais ao longo das direções coordenadas, uma de cada vez (Nocedal & Wright 1999).

O passo inicial do algoritmo é na direção de descida e as subseqüentes direções são encontradas a partir de

$$p_{i+1} = -g_{i+1} + \beta_i p_i,$$

onde $\beta_i = \dfrac{||g_{i+1}||^2}{||g_i||^2}$ e $g_i = g(x_i)$ representa o gradiente da função calculado no ponto x_i. Aqui, o subescrito i indexa as iterações. Este processo garante que será localizado o ponto mínimo de qualquer função quadrática de d argumentos em, no máximo, d iterações (Everitt 1987).

Para funções não-quadráticas, o processo realiza mais de d iterações e é requerido um teste de convergência. As direções p_i geradas são aquelas que correspondem à aproximação quadrática atual da função e a taxa de convergência depende da resposta a mudanças na aproximação quadrática local de iteração para iteração.

Fletcher & Reeves (1964) testaram o método com a função de Rosenbrock[1] em duas dimensões e encontraram que ele apresenta convergência lenta devido às sucessivas direções

[1]Esta é uma das funções de teste mais utilizadas, sendo definida por

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

quase paralelas que tornam muito próximos pontos resultantes de iterações sucessivas. Os autores adotaram uma solução para superar problemas com funções não-periódicas que nunca poderia reter a convergência quadrática do processo quando aplicada a funções periódicas. Tal solução consiste em voltar periodicamente à direção de descida $-g$ ao invés de p. Assim, o processo completo é periodicamente reiniciado no ponto atual, descartando toda experiência precedente, certa ou errada, que normalmente seria transmitida no cálculo de p. O processo permanece quadraticamente convergente desde que tais reinícios não sejam mais freqüentes do que a cada d iterações.

3.2.2 Método Newton-Raphson

Talvez o método mais conhecido de busca unidimensional de raízes de funções seja o de Newton, também conhecido como Newton-Raphson. Geometricamente, o método consiste em traçar a reta tangente à função no ponto x_i e tomar como x_{i+1} o ponto determinado pela intersecção desta reta com o eixo das abcissas, calculando o valor da função nesse ponto e traçando uma nova tangente. Encontra-se, assim, uma melhor aproximação da raiz a cada iteração. O processo pára quando é alcançada a precisão estipulada. Este método não está restrito a apenas uma dimensão e pode ser facilmente generalizado para múltiplas dimensões (Press et al. 1992).

O algoritmo de maximização de Newton-Raphson pode ser obtido a partir da expansão da função $f(\theta)$ em série de Taylor em torno de um ponto $\theta(k)$ até a segunda ordem, obtendo-se a seguinte aproximação quadrática:

$$f(\theta) \cong f(\theta(k)) + (\theta - \theta(k))^\top \nabla f(\theta(k)) + \frac{1}{2}(\theta - \theta(k))^\top H(\theta(k))(\theta - \theta(k)) = q(\theta),$$

onde $\nabla f(\theta(k))$ é o gradiente da função f avaliado em $\theta(k)$, $H(\theta(k))$ é a matriz hessiana avaliada no ponto $\theta(k)$ e θ pertence a uma vizinhança de $\theta(k)$. O próximo ponto da iteração, denotado por $\theta(k+1)$, é dado pelo máximo de $q(\theta)$. Com este fim, primeiro se determina o gradiente de $q(\theta)$, i.e.,

$$\nabla q(\theta) = \nabla f(\theta(k)) + H(\theta(k))(\theta - \theta(k));$$

faz-se, então, $\nabla q(\theta) = 0$, obtendo-se

$$H(\theta(k))(\theta - \theta(k)) = -\nabla f(\theta(k)).$$

Supondo que a inversa de $H(\theta(k))$ existe, tem-se que

$$\theta = \theta(k) - [H(\theta(k))]^{-1} \nabla f(\theta(k)),$$

θ sendo uma aproximação para o ponto maximizador da função f. Assim, aproximações para o ponto de máximo desta função são obtidas através da lei de recorrência

$$\theta(k+1) = \theta(k) - [H(\theta(k))]^{-1} \nabla f(\theta(k)).$$

Generalizando, tem-se

$$\theta(k+1) = \theta(k) - s(k) [H(\theta(k))]^{-1} \nabla f(\theta(k)),$$

onde $s(k)$ é um escalar determinado por um procedimento de busca linear a partir de $\theta(k)$ na direção de $-[H(\theta(k))]^{-1} \nabla f(\theta(k))$ de tal forma que $f(\theta(k))$ cresça ao longo desta direção. O aumento da função f é caracterizado pela desigualdade

$$\nabla^\top f(\theta(k))(\theta - \theta(k)) > 0 \Rightarrow -(\theta - \theta(k))^\top H(\theta(k))(\theta - \theta(k)) > 0.$$

Portanto, H deve ser negativa-definida para todo k. Porém, quando θ está longe do ponto de máximo, não há garantia de que a matriz hessiana satisfaça esta condição. Assim, o incremento $-s(k)\left[H(\theta(k))\right]^{-1}\nabla f(\theta(k))$ pode mover o ponto $\theta(k)$ para um ponto $\theta(k+1)$ no qual o valor da função é menor, podendo acarretar não convergência do algoritmo. Uma opção é utilizar o algoritmo BFGS (Broyden, Fletcher, Goldfarb & Shanno), que não sofre deste problema.

Os métodos do tipo Newton apresentam taxa rápida de convergência local, tipicamente quadrática. Quando uma vizinhança da solução é alcançada, freqüentemente ocorre a convergência com alta precisão em poucas iterações. Porém, longe do ponto ótimo, onde os termos de ordem superior na expansão de Taylor são importantes, o método pode fornecer soluções extremamente imprecisas. Este método tem tendência à não convergência global, pois pode convergir a diferentes soluções dependendo do ponto inicial. Para contornar este problema têm sido desenvolvidas várias modificações que combinam as propriedades de rápida convergência local do método Newton com estratégias globalmente convergentes que garantam progressos na direção da solução a cada iteração; ver Sherman (1978), Moré & Sorensen (1979), Goldfarb (1980), Denbo, Eisenstat & Steihaug (1982) e Schnabel & Eskow (1991), entre outros.

Um dos inconvenientes destes métodos é a necessidade do uso da matriz hessiana, pois o cálculo explícito da matriz de segundas derivadas é, muitas vezes, um processo computacionalmente custoso e convidativo ao erro. Para mais detalhes sobre este método de otimização, ver Nocedal & Wright (1999) e Press et al. (1992).

3.3 Métodos Quasi-Newton

Os métodos de métrica variável ou quasi-Newton derivam do método de Newton, tendo o mesmo objetivo: acumular informações das sucessivas minimizações em linha que levem ao ótimo exato de uma forma quadrática em d dimensões. A técnica dos métodos quasi-Newton difere da utilizada pelos métodos de gradiente conjugado na forma de armazenar e de atualizar essas informações acumuladas, pois, ao invés de requerer armazenamento de ordem d (o número de dimensões), requer-se armazenamento de uma matriz de tamanho $d \times d$ (Press et al. 1992).

Uma outra diferença entre os métodos de Newton e quasi-Newton reside no cálculo da inversa da matriz hessiana, que é aproximada por uma matriz simétrica e positiva-definida $Q(k)$ tal que

$$\lim_{k \to \infty} Q(k) = -H^{-1}.$$

A matriz $Q(k)$ é atualizada depois de cada passo para levar em conta o conhecimento adicional ganho durante a iteração. As atualizações usam o fato de que as mudanças no gradiente proporcionam informação sobre a segunda derivada da função ao longo da direção de busca (Nocedal & Wright 1999).

A matriz inicial mais comumente usada para $Q(k)$ é a matriz identidade de mesma ordem, pois ela é positiva-definida e simétrica, resultando, assim, em aproximações $Q(k)$ positivas-definidas e simétricas.

O método quasi-Newton mais utilizado é o *BFGS (Broyden, Fletcher, Goldfarb e Shanno)*,

cuja forma recursiva de atualização da aproximação da matriz hessiana é dada pela expressão

$$
\begin{aligned}
Q(k+1) = Q(k) &+ \frac{(\theta(k+1)-\theta(k)) \otimes ((\theta(k+1)-\theta(k)))}{(\theta(k+1)-\theta(k))^\top (\nabla f(\theta(k+1)) - \nabla f(\theta(k)))} \\
&- \frac{[Q(k)(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))] \otimes [Q(k)(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))]}{(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))^\top Q(k)(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))} \\
&+ (\nabla f(\theta(k+1)) - \nabla f(\theta(k)))^\top Q(k)(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))U \otimes U,
\end{aligned}
$$

onde o operador $\otimes$ denota produto de Kronecker e U é o vetor coluna dado por

$$
\begin{aligned}
U = &\frac{\theta(k+1)-\theta(k)}{(\theta(k+1)-\theta(k))^\top (\nabla f(\theta(k+1)) - \nabla f(\theta(k)))} \\
&- \frac{Q(k)(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))}{(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))^\top Q(k)(\nabla f(\theta(k+1)) - \nabla f(\theta(k)))}.
\end{aligned}
$$

Dessa forma, mesmo quando θ está longe do máximo, as matrizes $Q(k)$ garantem que os pontos se movem em direção ascendente. De forma semelhante ao método anterior, o esquema iterativo é

$$\theta(k+1) = \theta(k) + s(k)Q(\theta(k))\nabla f(\theta(k)),$$

onde $s(k)$ é o tamanho do passo. Embora o método de Newton convirja mais rapidamente (ou seja, quadraticamente), seu custo por iteração é maior devido à necessidade da solução de um sistema linear. Uma vantagem mais notável do método BFGS é que ele não requer o cálculo de segundas derivadas (Nocedal & Wright 1999).

Embora seja raro, pode acontecer que erros de arredondamento produzam uma matriz $Q(\theta(k))$ quase singular ou não positiva-definida. Isto pode ser sério, pois as direções de busca neste caso não seriam em descida e, adicionalmente, matrizes quase singulares tendem a induzir matrizes quase singulares subseqüentes. Para lidar com este inconveniente, têm sido implementadas variantes do método que consistem em construir uma aproximação de

H ao invés de H^{-1}. Aqui o truque consiste em armazenar, não esta aproximação, mas sua decomposição triangular ou decomposição de Cholesky, que pode ser reorganizada de forma a garantir que a matriz permaneça não-singular e positiva-definida mesmo na presença de erros de arredondamento (Press et al. 1992).

3.4 Método Simplex

Este método está baseado nos simplexes, que são figuras geométricas em d dimensões definidas por $d+1$ vértices. Por exemplo, um simplex em duas dimensões é um triângulo, em três dimensões é um tetraedro e assim sucessivamente. O procedimento de otimização deste método começa pela escolha dos $d+1$ pontos onde será feita a avaliação da resposta, o objetivo sendo forçar o simplex a mover-se para a região de resposta ótima através de operações de reflexão, expansão e contração. Este algoritmo não precisa das derivadas da função e tipicamente não é eficiente em termos do número de avaliações da função até convergência.

Nelder & Mead (1965) desenvolveram o algoritmo conhecido como *simplex downhill*, que considera a minimização de uma função de d variáveis, sem restrições. O método pode ser descrito como a seguir. Sejam $P_0, P_1, \ldots, P_d$ os $(d+1)$ pontos no espaço d-dimensional definindo o simplex atual. Denota-se por y_i o valor da função em P_i e define-se h como o subíndice tal que $y_h = \max_i(y_i)$ e l como o subíndice tal que $y_l = \min_i(y_i)$. Ainda, define-se $\bar{P}$ como o centróide dos pontos com $i \neq h$ e $[P_i P_j]$ como a distância entre P_i e P_j. Em cada etapa do processo, P_h é trocado por um novo ponto usando três operações:

- Reflexão de P_h: Denotada por P^*, suas coordenadas são definidas pela relação

$$P^* = (1 + \alpha)\bar{P} - \alpha P_h,$$

em que α (coeficiente de reflexão) é uma constante positiva. Assim, P^* está sobre a reta que une P_h e $\bar{P}$ com $[P^*\bar{P}] = \alpha[P_h\bar{P}]$. Se y^* estiver entre y_h e y_l, então P_h é trocado por P^* e o processo é reiniciado com o novo simplex.

- Expansão de P_h: Se $y^* < y_l$, i.e., se a operação de reflexão produz um novo ponto mínimo, então expande-se P^* a P^{**} pela relação

$$P^{**} = \gamma P^* + (1 - \gamma)\bar{P}.$$

O coeficiente de expansão γ, que é maior do que 1, é a razão entre as distâncias $[P^{**}\bar{P}]$ e $[P^*\bar{P}]$. Se $y^{**} < y_l$, troca-se P_h por P^{**} e o processo é reiniciado; se $y^{**} > y_l$, então a expansão é falha e deve-se trocar P_h por P^* antes de recomeçar o processo.

- Contração de P_h: Se na reflexão de P a P^* se encontra $y^* > y_i$ para todo $i \neq h$, i.e., que a troca de P por P^* resulta em um y^* máximo, então define-se um novo P_h sendo igual ao P_h anterior ou a P^* (qualquer um deles possui menor valor y). Então forma-se

$$P^{**} = \beta P_h + (1 - \beta)\bar{P}.$$

O coeficiente de contração β está entre 0 e 1 e é a razão das distâncias $[P^{**}\bar{P}]$ e $[P\bar{P}]$. Assim, aceita-se P^{**} para P_h e o processo é reiniciado, a não ser que $y^{**} > \min(y_h, y^*)$, i.e., o ponto depois da operação de contração é "pior" do que o "melhor" entre P_h e P^*. Quando ocorre tal falha na contração, todos os P_i são trocados por $(P_i + P_l)/2$ e o processo é reiniciado.

Finalmente, o critério de parada do algoritmo consiste em comparar o "erro padrão" dos y's, $\sqrt{\{\sum(y_i - \bar{y})^2/n\}}$, com um valor pré-fixado. O processo pára quando este valor é maior que o erro padrão. O sucesso deste critério depende do simplex não chegar a ser muito pequeno em relação à curvatura da superfície até que o mínimo seja alcançado. A motivação por trás deste critério deve-se a que em problemas estatísticos em que o objetivo é encontrar o mínimo de uma superfície de verossimilhança negativa (ou de uma superfície de soma de quadrados) a curvatura próxima do mínimo provê informação sobre os parâmetros desconhecidos. Se a curvatura for leve, a variância amostral das estimativas será grande e não tem sentido encontrar as coordenadas do mínimo com muita precisão, ao passo que se a curvatura for acentuada, há justificativa para se procurar o ponto de mínimo com maior exatidão (Nelder & Mead 1965).

CAPÍTULO 4

Avaliação Numérica

4.1 Introdução

Através de simulações de Monte Carlo foram avaliados os desempenhos de diferentes métodos de otimização não-linear no que tange à maximização da função de log-verossimilhança do modelo de regressão beta proposto por Ferrari & Cribari-Neto (2004). No estudo de simulação foram comparadas as estimativas obtidas no processo de otimização através dos métodos BFGS, Newton, simplex e gradiente conjugado (CG), descritos no Capítulo 3, sob diferentes cenários. Outras variantes do método de Newton foram avaliadas, como, por exemplo, os métodos *scoring* de Fisher (que usa o valor esperado da matriz de segundas derivadas ao invés da matriz hessiana), steepest descent (que usa a matriz identidade ao invés da matriz hessiana) e BHHH (Berndt, Hall, Hall & Hausman 1974), mas os

resultados não diferiram de forma clara dos obtidos a partir do método de Newton-Raphson e, portanto, não são apresentados.

Os dados foram gerados do modelo de regressão beta dado por

$$g(\mu_t) = \beta_0 + \beta_1 x_t, \quad t = 1, \ldots, n, \tag{4.1}$$

onde $g(\cdot)$ é a função de ligação logit. Os valores da variável regressora x_t foram obtidos como realizações independentes da distribuição normal padrão. A matriz de variáveis regressoras X permaneceu constante ao longo do experimento e para cada réplica de Monte Carlo foi gerada uma amostra aleatória da variável resposta $\mathbf{y} = (y_1, \ldots, y_n)^\top$, onde y_t é obtido da distribuição beta apresentada em (2.2), i.e., $y_t \sim \mathcal{B}(\mu_t, \phi)$, $t = 1, \ldots, n$. O parâmetro de locação, μ_t, é dado por

$$\mu_t = \frac{\exp(\beta_0 + \beta_1 x_t)}{1 + \exp(\beta_0 + \beta_1 x_t)}.$$

Os valores considerados para os parâmetros de intercepto e inclinação foram $\beta_0 = 1.0$ e $\beta_1 = 0.1$, respectivamente; o valor do parâmetro de precisão foi fixado em $\phi = 35$ e os tamanhos amostrais utilizados foram $n = 20, 60, 100, 200$ e 500. Para cada tamanho amostral foram calculadas as medidas descritivas: média e variância. Adicionalmente foi calculado o critério de avaliação erro absoluto percentual médio (MAPE) nas 1000 réplicas de Monte Carlo realizadas.

4.2 Ambiente Computacional

Todos os experimentos foram conduzidos sob o sistema operacional Microsoft Windows 2000 Professional, em um computador com processador Pentium IV de 2.40 GHz e 1.0

Gb em memória RAM. A simulação foi realizada através de procedimentos implementados na linguagem de programação `Ox`, em sua versão `4.02` (distribuída gratuitamente para uso acadêmico e disponível no site `http://www.doornik.com`), e no software `R` em sua versão `2.1.1` (disponível gratuitamente em `http://www.r-project.org`).

As avaliações numéricas dos métodos de otimização não-linear foram executadas utilizando as rotinas de otimização `MaxBFGS`, `MaxNewton` e `MaxSimplex` do sistema `Ox`; os resultados relativos ao método CG foram obtidos a partir da função `optim` do software `R`; a versão usada desse método foi aquela desenvolvida por Fletcher & Reeves (1964). Os métodos de tipo gradiente e de Newton foram utilizados com gradiente analítico da função de log-verossimilhança e, quando necessário, com matriz hessiana numérica. Para maiores detalhes sobre estas rotinas de otimização, ver Doornik & Ooms (2005), no caso do `Ox`, e Venables & Smith (2004), no caso do `R`.

4.3 Resultados

Inicialmente foram avaliadas as precisões dos diferentes métodos na estimação dos parâmetros do modelo (4.1). Os valores iniciais (β_{0_0}, β_{1_0} e ϕ_0) para o processo iterativo da estimação dos parâmetros foram obtidos como descrito na Seção 2.3 do Capítulo 2. Os valores iniciais médios obtidos para cada parâmetro com diferentes tamanhos de amostra nas 1000 réplicas de Monte Carlo realizadas, são apresentados na Tabela 4.1. Os valores β_{0_0} e β_{1_0} estão próximos dos verdadeiros valores dos parâmetros β_0 e β_1, independentemente do tamanho de amostra, enquanto que o valor do ϕ_0 não se encontra muito próximo do valor verdadeiro.

n	$\beta_{\mathbf{0_0}}$	$\beta_{\mathbf{1_0}}$	$\phi_{\mathbf{0}}$
20	0.89446	0.12707	31.44047
60	1.05745	0.10087	23.63574
100	0.99091	0.12537	29.27570
200	1.08458	0.09968	29.06153
500	1.01183	0.10887	32.45599

Tabela 4.1: Valores iniciais na estimação dos parâmetros $\beta_0 = 1.0$, $\beta_1 = 0.1$ e $\phi = 35$.

Em resumo, o valor inicial para a estimação do parâmetro de locação (μ_t) está próximo do valor verdadeiro, diferentemente do que ocorre com o parâmetro de precisão do modelo (ϕ). Isto pode produzir alguma desvantagem para a eficiência dos métodos no que tange à estimação do parâmetro ϕ, uma vez que, em pontos afastados do máximo, a aproximação quadrática da função de verossimilhança pode não ser precisa, fazendo com que o algoritmo necessite de maior número de iterações, levando assim maior tempo para atingir o ponto ótimo.

A Tabela 4.2 apresenta os resultados obtidos para o parâmetro de intercepto do modelo. Observa-se que todos os métodos proporcionaram valores semelhantes no que tange às estimativas. Os métodos BFGS e Newton forneceram os mesmos resultados para as quantidades calculadas e sempre apresentaram a menor variância das estimativas. O método simplex foi o que apresentou, em geral, maior variabilidade das estimativas, mesmo para tamanhos de amostra grandes. A diferença nas medidas de viés e variância entre os resultados fornecidos pelos métodos CG e BFGS (assim como o método de Newton) tornou-se menor na medida em que o tamanho de amostra aumentou.

No referente à taxa de erro (MAPE), os métodos BFGS e Newton forneceram as menores taxas para tamanhos amostrais pequenos, enquanto que para tamanhos grandes o método

simplex proporcionou as menores taxas, seguido pelos métodos BFGS e Newton. Por exemplo, para $n = 200$ o MAPE do método simplex foi de 4.32% enquanto que para o método BFGS esta medida foi de 4.34%, indicando assim que os métodos BFGS, Newton e simplex podem ser usados indistintamente na estimação do parâmetro β_0, no que se refere a este critério, quando o tamanho amostral é suficientemente grande. As maiores taxas de erro foram exibidas pelo método CG para todos os tamanhos de amostra, sendo que, para tamanhos de amostra grandes, estas têm magnitude semelhante às dos métodos BFGS e Newton.

n	**Medida**	**BFGS**	**Newton**	**simplex**	**CG**
20	Média	1.0032	1.0032	1.0028	1.0057
	Variância	0.0275	0.0275	0.0279	0.0360
	MAPE	13.1400	13.1400	13.1390	14.9989
60	Média	1.0002	1.0002	1.0003	1.0025
	Variância	0.0108	0.0108	0.0108	0.0108
	MAPE	8.2950	8.2950	8.2980	8.3636
100	Média	1.0017	1.0017	0.9999	0.9996
	Variância	0.0059	0.0059	0.0075	0.0064
	MAPE	6.1120	6.1120	6.9130	6.3739
200	Média	1.0001	1.0001	0.9950	1.0003
	Variância	0.0030	0.0030	0.0080	0.0032
	MAPE	4.3420	4.3420	4.3240	4.5366
500	Média	0.9999	0.9999	0.9676	0.9995
	Variância	0.0013	0.0013	0.0681	0.0013
	MAPE	2.8380	2.8380	2.6350	2.8825

Tabela 4.2: Resultados da simulação para as estimativas de máxima verossimilhança do parâmetro $\beta_0 = 1.0$.

Realizando a mesma análise para o parâmetro de inclinação do modelo (β_1), achou-se que os resultados obtidos foram análogos aos do parâmetro de intercepto. Todos os métodos estudados forneceram estimativas semelhantes, apresentando menores variâncias os métodos BFGS e Newton, como observado na Tabela 4.3. Para tamanhos amostrais pequenos,

o método simplex apresentou variância igual às dos métodos BFGS e Newton, enquanto que para os maiores tamanhos de amostra ele apresentou a maior variabilidade. Observa-se também que, embora o método CG tenha apresentado a mesma variância nas estimativas dos métodos BFGS e Newton para tamanhos de amostra suficientemente grandes, suas taxas de erro tenderam a ser as maiores. Por exemplo, para $n = 60$ seu MAPE foi de 15.86% enquanto que para os outros métodos o valor desse critério esteve próximo de 13.92%; para $n = 100$, seu MAPE foi de 13.47%, os outros métodos apresentando valor do MAPE em torno de 11.19%. Este fato coloca novamente o método CG em desvantagem em relação aos demais.

n	**Medida**	**BFGS**	**Newton**	**simplex**	**CG**
20	Média	0.1006	0.1006	0.1006	0.1004
	Variância	0.0012	0.0012	0.0012	0.0014
	MAPE	26.9970	26.9970	26.9780	29.2827
60	Média	0.1001	0.1001	0.1001	0.0999
	Variância	0.0003	0.0003	0.0003	0.0004
	MAPE	13.9190	13.9190	13.9250	15.8608
100	Média	0.1000	0.1000	0.1003	0.1004
	Variância	0.0002	0.0002	0.0003	0.0002
	MAPE	11.1880	11.1880	13.4680	12.2886
200	Média	0.1001	0.1001	0.0996	0.1001
	Variância	0.0001	0.0001	0.0005	0.0001
	MAPE	8.2930	8.2930	8.2560	8.6677
500	Média	0.1000	0.1000	0.0928	0.1001
	Variância	0.0001	0.0001	0.0007	0.0001
	MAPE	5.4600	5.4600	5.0640	5.4430

Tabela 4.3: Resultados da simulação para as estimativas de máxima verossimilhança do parâmetro $\beta_1 = 0.1$.

Observa-se na Tabela 4.4 que, para tamanhos de amostra pequenos, existe viés significativo nas estimativas do parâmetro de precisão (ϕ) e que a variância de tais estimativas é consideravelmente alta, tornando-se menor rapidamente com o aumento do número de ob-

servações, como esperado. Os valores estimados fornecidos pelos diferentes métodos foram muito homogêneos, sendo o método simplex aquele que apresentou sempre a estimativa mais próxima do verdadeiro valor do parâmetro. Verifica-se que os métodos BFGS, Newton e simplex apresentam valores parecidos no critério MAPE; novamente, o método CG foi o menos eficiente, com as maiores taxas de erro para todos os tamanhos amostrais. Em geral, o método simplex apresentou as menores taxas de erro, indicando que possivelmente uma opção razoável para não contribuir ao aumento do viés na estimação desse parâmetro é o uso de um método que não dependa da informação relativa ao gradiente da função.

n	**Medida**	**BFGS**	**Newton**	**simplex**	**CG**
20	Média	43.9741	43.9741	43.9715	43.9905
	Variância	273.7639	273.7639	274.2863	270.0874
	MAPE	37.0300	37.0300	37.0020	37.1427
60	Média	37.5441	37.5441	37.4428	37.5192
	Variância	51.7303	51.7303	51.7413	49.5842
	MAPE	16.3900	16.3900	16.3990	16.6759
100	Média	36.5494	36.5494	36.4109	36.4984
	Variância	27.6131	27.6131	27.9915	27.9175
	MAPE	12.1140	12.1140	11.9490	12.1407
200	Média	35.7115	35.7115	35.5278	35.7096
	Variância	12.8998	12.8998	19.3014	18.2372
	MAPE	8.2040	8.2040	8.1610	8.3627
500	Média	35.2698	35.2698	35.2085	35.2424
	Variância	4.9738	4.9738	7.9740	7.9975
	MAPE	5.0820	5.0820	4.7210	5.8254

Tabela 4.4: Resultados da simulação para as estimativas de máxima verossimilhança do parâmetro $\phi = 35$.

Em geral, pode-se deduzir das Tabelas 4.2, 4.3 e 4.4 que, em princípio, os métodos BFGS, Newton e simplex são competitivos no que tange à obtenção das estimativas de máxima verossimilhança dos parâmetros do modelo de regressão beta. Os resultados da simulação indicam que qualquer um desses métodos pode ser usado na estimação do parâmetro de

locação, enquanto que não há suficiente clareza sobre qual método é o mais adequado para a estimação do parâmetro de precisão. O método gradiente conjugado não é recomendável, pois é propenso a alcançar as maiores taxas de erro. O fato de que os métodos de Newton e BFGS sempre fornecem as mesmas estimativas, assim como a mesma variabilidade e taxas de erro, indica que não está havendo dificuldade no que concerne à inversão da matriz hessiana no método de Newton.

Para os valores de n considerados, observou-se que a taxa MAPE diminui aproximadamente em 50% com o aumento do tamanho amostral, independentemente do método usado. Ainda, esta taxa foi sempre maior na estimação de ϕ, evidenciando a dificuldade dos métodos para estimar com precisão esse parâmetro.

Um segundo experimento de simulação foi realizado, desta vez com um modelo com duas covariáveis. As conclusões extraídas foram as mesmas que no caso de uma covariável, i.e., todos os métodos forneceram estimativas semelhantes para o parâmetro de locação, havendo uma certa superioridade dos métodos que usam a informação da matriz hessiana além do gradiente da função. No referente ao parâmetro de precisão, houve maior dificuldade na determinação do método que apresenta o melhor desempenho de acordo com os critérios de avaliação. Independentemente do tamanho de amostra, o método simplex foi o que tipicamente apresentou as menores taxas de erro, apesar de ter fornecido as estimativas com maior variabilidade.

Visando verificar se o viés significativo nas estimativas de ϕ está relacionado com o ponto inicial, foram comparados os resultados fornecidos por cada método para diferentes valores iniciais do parâmetro ϕ, a saber: o valor inicial descrito na Seção 2.3 ($\phi_{0_{FC}}$), o verdadeiro

valor do parâmetro ($\phi_0 = 35$), um valor menor e outro maior do que o verdadeiro valor do parâmetro ($\phi_0 = 15$ e $\phi_0 = 70$, respectivamente). Observa-se na Tabela 4.5 que não há grande diferença nas estimativas fornecidas pelos métodos BGFS e Newton. Nesse caso, os resultados apresentam insensibilidade ao ponto inicial e o viés diminui com o aumento do n, como esperado. O método simplex fornece estimativas e valores do critério de avaliação semelhantes aos obtidos com os métodos BFGS e Newton. Para este método são observadas leves diferenças nas estimativas causadas pelos diferentes pontos iniciais. O método CG fornece pontos diferentes dependendo do valor inicial utilizado; adicionalmente, a variabilidade das suas estimativas tende a ser menor quando o processo começa em um ponto próximo do verdadeiro valor do parâmetro, como é o caso do ponto $\phi_{0_{FC}}$.

Como esperado, existe uma tendência dos métodos a apresentar menores taxas de erro quando o ponto inicial coincide com o verdadeiro valor do parâmetro. Novamente, as maiores taxas de erro foram as do método CG e as estimativas mais próximas do valor verdadeiro foram as fornecidas pelo método simplex. Observa-se também que a variabilidade é grande para pequenos tamanhos amostrais, diminuindo rapidamente com o aumento do número de observações; igualmente, os valores dos critérios de avaliação diminuem rapidamente. O tempo gasto por cada método no processo de maximização foi também usado como critério de avaliação. Observa-se na Tabela 4.6 que os métodos simplex e CG precisaram sempre de mais tempo para convergir, sendo que o método simplex foi mais lento. Mais importante ainda é observar que, embora as estimativas fornecidas pelos métodos BFGS e Newton sejam iguais, este último precisou de menos tempo para realizar o processo de otimização.

		BFGS			**Newton**			**simplex**			**CG**		
n	**Medida**	$\phi_0 = 15$	$\phi_0 = 35$	$\phi_0 = 70$	$\phi_0 = 15$	$\phi_0 = 35$	$\phi_0 = 70$	$\phi_0 = 15$	$\phi_0 = 35$	$\phi_0 = 70$	$\phi_0 = 15$	$\phi_0 = 35$	$\phi_0 = 70$
20	Média	43.9741	43.9741	43.9741	43.9741	43.9741	43.9741	43.9626	43.9572	43.9568	43.8581	43.8101	43.9412
	Variância	273.7639	273.7639	273.7641	273.7639	273.7639	273.7641	274.0156	274.1241	274.1579	280.0874	280.2947	280.2954
	MAPE	37.0300	37.0300	37.0300	37.0300	37.0300	37.0300	37.0170	37.0100	37.0100	37.5080	37.5240	37.5865
60	Média	37.5441	37.5441	37.5441	37.5441	37.5441	37.5441	37.5482	37.5399	37.5405	37.5271	37.5804	37.5541
	Variância	51.7303	51.7304	51.7303	51.7304	51.7303	51.7303	52.0045	51.7003	51.7582	51.1975	50.0001	49.7303
	MAPE	16.3900	16.3900	16.3900	16.3900	16.3900	16.3900	16.4330	16.3840	16.3960	16.4357	16.2255	16.0840
100	Média	36.5494	36.5494	36.5494	36.5494	36.5494	36.5494	36.2801	36.5382	36.5513	36.6143	36.4415	36.6081
	Variância	28.2322	27.6131	27.6131	28.2322	27.6131	27.6131	33.3614	27.6620	27.7911	28.0567	27.9965	27.9220
	MAPE	12.1140	12.1140	12.1140	12.1140	12.1140	12.1140	12.0730	12.1280	12.1390	12.2755	12.1632	12.0235
200	Média	35.7115	35.7115	35.7115	35.7115	35.7115	35.7115	35.5183	35.4999	35.5089	35.6585	35.6376	35.6237
	Variância	12.8998	12.8998	12.8998	12.8998	12.8998	12.8998	19.4357	20.3108	19.7923	20.4598	22.4623	20.1486
	MAPE	8.2040	8.2040	8.2040	8.2040	8.2040	8.2040	8.1600	8.1610	8.1550	8.2705	8.2176	8.2705
500	Média	35.2698	35.2698	35.2698	35.2698	35.2698	35.2698	35.3605	35.2382	35.4480	35.2935	35.2974	35.2548
	Variância	4.9738	4.9738	4.9738	4.9738	4.9738	4.9738	8.5441	8.9082	8.6196	3.5621	7.0165	5.4462
	MAPE	5.0820	5.0820	5.0820	5.0820	5.0820	5.0820	4.7440	4.7220	4.7400	5.0585	5.0276	4.9225

Tabela 4.5: Resultados da simulação para as estimativas de máxima verossimilhança do parâmetro $\phi = 35$ com diferentes valores iniciais do parâmetro de precisão.

Todos os métodos gastaram menos tempo de execução quando o valor inicial foi aquele apresentado na Tabela 4.1 ou menor do que o valor verdadeiro; adicionalmente, evidenciou-se maior dificuldade quando o valor inicial utilizado foi maior do que o valor verdadeiro do parâmetro.

n	**Método**	$\phi_{\mathbf{0_{FC}}}$	$\phi_{\mathbf{0}} = \mathbf{35}$	$\phi_{\mathbf{0}} = \mathbf{15}$	$\phi_{\mathbf{0}} = \mathbf{70}$
20	$BFGS$	0 : 01 : 17	0 : 01 : 17	0 : 01 : 15	0 : 01 : 21
	$Newton$	0 : 00 : 48	0 : 00 : 50	0 : 01 : 00	0 : 00 : 58
	$simplex$	0 : 07 : 10	0 : 08 : 50	0 : 05 : 44	0 : 06 : 43
	CG	0 : 05 : 20	0 : 06 : 54	0 : 03 : 46	0 : 05 : 27
60	$BFGS$	0 : 03 : 59	0 : 05 : 58	0 : 04 : 25	0 : 04 : 08
	$Newton$	0 : 02 : 21	0 : 02 : 22	0 : 03 : 09	0 : 02 : 57
	$simplex$	0 : 23 : 06	0 : 22 : 00	0 : 19 : 39	0 : 20 : 38
	CG	0 : 15 : 47	0 : 13 : 25	0 : 10 : 05	0 : 08 : 25
100	$BFGS$	0 : 06 : 49	0 : 04 : 54	0 : 08 : 16	0 : 07 : 47
	$Newton$	0 : 03 : 57	0 : 03 : 51	0 : 03 : 14	0 : 04 : 46
	$simplex$	0 : 34 : 13	0 : 40 : 34	0 : 31 : 09	1 : 10 : 23
	CG	0 : 30 : 18	0 : 34 : 23	0 : 25 : 06	0 : 50 : 36
200	$BFGS$	0 : 15 : 20	0 : 13 : 38	1 : 00 : 48	1 : 05 : 57
	$Newton$	0 : 09 : 05	0 : 08 : 02	0 : 42 : 36	0 : 43 : 21
	$simplex$	1 : 14 : 49	1 : 19 : 40	1 : 16 : 36	3 : 35 : 50
	CG	1 : 00 : 02	1 : 01 : 56	1 : 00 : 14	2 : 45 : 35
500	$BFGS$	5 : 21 : 10	5 : 31 : 19	6 : 22 : 38	6 : 37 : 29
	$Newton$	2 : 44 : 40	2 : 50 : 18	5 : 17 : 50	4 : 54 : 08
	$simplex$	12 : 01 : 16	8 : 54 : 17	7 : 55 : 41	20 : 49 : 46
	CG	10 : 52 : 46	6 : 35 : 42	6 : 05 : 23	18 : 58 : 32

Tabela 4.6: Tempo gasto pelos métodos no processo de maximização para diferentes valores iniciais do parâmetro de precisão. (Formato: hh:mm:ss)

Embora o método simplex conduza às estimativas mais precisas quando o ponto inicial está próximo do valor verdadeiro, este método não é computacionalmente eficiente, pois é muito lento, sendo que este comportamento piora quando o tamanho amostral aumenta. O método CG apresenta convergência lenta nas maiores amostras. Nota-se a eficiência do método de Newton na maximização da log-verossimilhança do modelo de regressão (4.1).

Um outro interesse reside em determinar a robustez dos métodos à presença de pontos de alavanca[1] nos dados. Com este fim, foram gerados $n-1$ valores da variável regressora x_t a partir da distribuição normal padrão e o n-ésimo ponto de cada amostra foi fixado em $2.5, 3.0, 4.0, 5.0, 6.0$ ou 7.0. Neste caso, o critério de comparação foi o número de iterações realizadas por cada método para alcançar o ponto ótimo. (Pontos de alavanca induzem viés nas estimativas e, portanto, não é possível determinar até que ponto os desvios são causados pelo ponto de alavanca e até que ponto são devidos ao processo de otimização.)

A Tabela 4.7 apresenta o número médio de iterações de cada método nas 1000 réplicas de Monte Carlo realizadas, para os tamanhos amostrais $n = 20$ e $n = 100$. Observa-se que o método mais eficiente é o de Newton seguido de longe pelo BFGS; o método que precisa de maior número de iterações é o simplex, com mais de 100 iterações em média. Em todos os métodos há tendência de aumento do número de iterações quando a magnitude do ponto de alavanca aumenta. O método que permaneceu mais estável às perturbações nos dados foi o de Newton.

		Magnitude do ponto de alavanca					
Método	n	**2.5**	**3.0**	**4.0**	**5.0**	**6.0**	**7.0**
BFGS	20	12	12	13	14	16	21
	100	10	11	11	12	13	14
Newton	20	3	3	4	4	5	6
	100	3	3	3	3	4	4
simplex	20	141	144	147	148	151	157
	100	128	133	137	144	147	145
CG	20	40	39	39	52	60	76
	100	45	46	55	50	52	53

Tabela 4.7: Número médio de iterações até convergência dos métodos na presença de pontos de alavanca.

[1]Pontos de alavanca são observações com valores atípicos das covariáveis e podem afetar as propriedades dos estimadores em amostras finitas, para maiores detalhes, ver Montgomery, Peck & Vining (2001).

Também é importante verificar as taxas de convergência dos métodos. Os métodos BFGS e de Newton convergiram em todos os casos, independentemente da magnitude do ponto de alavanca e do tamanho amostral. A Tabela 4.8 reporta as taxas de convergência dos métodos simplex e CG, que foram os únicos métodos afetados nas suas propriedades de convergência pela existência dos pontos de alavanca. Nota-se que o método simplex foi o mais robusto, pois a redução da sua taxa de convergência foi baixa comparada com a da taxa do método CG.

		Magnitude do ponto de alavanca					
Método	n	**2.5**	**3.0**	**4.0**	**5.0**	**6.0**	**7.0**
simplex	20	100.0	100.0	100.0	100.0	95.2	77.3
	100	99.4	99.6	99.3	99.7	99.5	92.9
CG	20	81.1	88.7	83.5	81.8	64.9	44.8
	100	62.5	55.8	61.9	62.2	63.7	63.1

Tabela 4.8: Taxas de convergência na presença de pontos de alavanca.

Finalmente, foi avaliada a robustez dos métodos estudados à existência de correlação acentuada entre as variáveis regressoras. Com este fim, foram geradas amostras de tamanho n da distribuição normal bivariada com vetor de médias $\mu = (0, 0)$ e matriz de covariâncias

$$\Sigma = \begin{pmatrix} 1 & \sigma_{x_1x_2} \\ \sigma_{x_2x_1} & 1 \end{pmatrix},$$

onde $\sigma_{x_1x_2} = \sigma_{x_2x_1} = 0.0, 0.1, 0.3, 0.5, 0.7$ ou 0.9. A estrutura do modelo empregado neste caso é dada por

$$g(\mu_t) = \beta_0 + \beta_1 x_{t1} + \beta_2 x_{t2}, \tag{4.2}$$

onde $g(\cdot)$ é a função de ligação logit e os valores dos parâmetros foram fixados em $\beta_0 = 1.0$, $\beta_1 = 0.1$, $\beta_2 = 0.5$ e $\phi = 35$.

Foram medidos os tempos gastos por cada método na maximização. Os resultados se encontram na Tabela 4.9. Nota-se que o método de Newton continua sendo o mais rápido e que o grau de correlação não afeta o processo de otimização dos métodos BFGS e Newton, pois os tempos de execução permanecem estáveis para todos os graus de correlação. No que diz respeito ao método simplex, para os maiores tamanhos amostrais há leve tendência de aumento do tempo gasto na otimização quando o grau de correlação é elevado.

Em termos gerais, observou-se que o método de Newton é uma ótima opção para o processo de maximização da função de log-verossimilhança do modelo de regressão beta, devido a sua rapidez, robustez e precisão para atingir o ponto ótimo. O método BFGS é competitivo também, dadas as precisões das suas estimativas, sendo, porém, sempre mais lento do que o método de Newton. Esta pode ser uma indicação de que para o caso específico da função de log-verossimilhança do modelo de regressão beta, a matriz hessiana não é de difícil inversão e está geralmente distante da singularidade.

Os métodos BFGS e de Newton se mostraram robustos a condições comumente verificadas na prática estatística, como elevada correlação entre variáveis regressoras e existência de pontos de alavanca nos dados. Embora o método simplex tenha sido tão eficiente quanto os dois métodos anteriormente mencionados, ele sempre necessita de mais iterações para atingir o ponto ótimo. Adicionalmente, quando submetido a algumas condições adversas, como a presença de pontos de alavanca, a taxa de convergência foi afetada. No caso do método CG, verificou-se que não é recomendável seu uso na obtenção de estimativas pontuais no modelo de regressão beta, pois sua precisão é menor do que as dos outros métodos; verificou-se que há uma baixa taxa de convergência quando os dados contêm pontos de alavanca.

n	**Método**	$\rho = \mathbf{0}$	$\rho = \mathbf{0.1}$	$\rho = \mathbf{0.3}$	$\rho = \mathbf{0.5}$	$\rho = \mathbf{0.7}$	$\rho = \mathbf{0.9}$
20	*BFGS*	00 : 00 : 15.76	00 : 00 : 15.61	00 : 00 : 15.18	00 : 00 : 15.34	00 : 00 : 16.57	00 : 00 : 17.56
	Newton	00 : 00 : 12.34	00 : 00 : 13.65	00 : 00 : 13.00	00 : 00 : 13.37	00 : 00 : 14.28	00 : 00 : 14.59
	simplex	00 : 01 : 12.76	00 : 01 : 22.79	00 : 01 : 08.64	00 : 01 : 08.81	00 : 01 : 22.53	00 : 01 : 23.40
	CG	00 : 55 : 12.00	00 : 01 : 00.00	00 : 45 : 48.00	00 : 48 : 48.00	00 : 36 : 00.00	00 : 43 : 12.00
60	*BFGS*	00 : 00 : 36.89	00 : 00 : 37.50	00 : 00 : 38.17	00 : 00 : 37.23	00 : 00 : 39.72	00 : 00 : 39.97
	Newton	00 : 00 : 30.43	00 : 00 : 29.34	00 : 00 : 30.67	00 : 00 : 31.22	00 : 00 : 27.82	00 : 00 : 31.95
	simplex	00 : 04 : 29.76	00 : 03 : 14.15	00 : 03 : 02.56	00 : 03 : 01.73	00 : 03 : 30.57	00 : 03 : 12.87
	CG	00 : 12 : 00.00	00 : 28 : 00.00	00 : 21 : 36.00	00 : 26 : 24.00	00 : 33 : 36.00	00 : 35 : 36.00
100	*BFGS*	00 : 01 : 12.45	00 : 01 : 09.39	00 : 01 : 10.53	00 : 01 : 14.67	00 : 01 : 16.07	00 : 01 : 13.45
	Newton	00 : 00 : 59.37	00 : 00 : 54.84	00 : 00 : 58.81	00 : 00 : 54.14	00 : 01 : 04.28	00 : 01 : 00.51
	simplex	00 : 07 : 20.78	00 : 06 : 07.95	00 : 07 : 45.80	00 : 08 : 25.95	00 : 12 : 32.40	00 : 08 : 10.39
	CG	00 : 43 : 00.00	00 : 40 : 48.00	00 : 38 : 24.00	00 : 50 : 24.00	00 : 37 : 12.00	00 : 38 : 24.00
200	*BFGS*	00 : 03 : 39.90	00 : 03 : 51.06	00 : 03 : 57.42	00 : 04 : 08.86	00 : 04 : 12.53	00 : 04 : 10.97
	Newton	00 : 03 : 15.75	00 : 03 : 22.75	00 : 03 : 20.22	00 : 03 : 26.67	00 : 02 : 59.62	00 : 03 : 20.75
	simplex	00 : 26 : 32.70	00 : 24 : 45.00	00 : 40 : 23.00	00 : 44 : 42.30	01 : 49 : 03.56	02 : 09 : 12.89
	CG	01 : 15 : 12.00	01 : 40 : 48.00	01 : 38 : 24.00	01 : 50 : 24.00	01 : 48 : 00.00	01 : 28 : 48.00
500	*BFGS*	00 : 38 : 13.10	00 : 38 : 12.70	00 : 37 : 32.90	00 : 39 : 36.30	00 : 37 : 23.20	00 : 39 : 59.50
	Newton	00 : 35 : 06.80	00 : 33 : 25.70	00 : 33 : 58.80	00 : 34 : 06.50	00 : 35 : 40.50	00 : 35 : 10.90
	simplex	07 : 25 : 43.73	07 : 03 : 32.67	14 : 28 : 53.00	20 : 10 : 40.00	18 : 39 : 46.00	22 : 19 : 55.00
	CG	03 : 43 : 12.00	02 : 59 : 36.00	03 : 52 : 48.00	02 : 24 : 00.00	05 : 58 : 55.00	04 : 38 : 24.00

Tabela 4.9: Tempo gasto no processo de maximização na presença de correlação entre as covariáveis (Formato: hh:mm:ss.ms)

Pode-se, então, concluir que os métodos que apresentaram melhor desempenho na maximização da função de log-verossimilhança do modelo de regressão beta foram BFGS e Newton. Quando o tamanho amostral é grande, pode-se usar o método de Newton, por sua rapidez no processo de otimização.

CAPÍTULO 5

Aplicações

5.1 Introdução

Neste capítulo são apresentadas e discutidas duas aplicações a dados reais do modelo de regressão beta descrito no Capítulo 2. A primeira aplicação utiliza os dados apresentados em Prater (1956), em que o interesse recai sobre a modelagem da proporção de petróleo cru convertido em gasolina após destilação e fracionamento. A segunda aplicação emprega os dados analisados por Smithson & Verkuilen (2006). Estes autores consideram a contribuição relativa do quociente intelectual (QI) não-verbal e a condição de dislexia ou não-dislexia para a distribuição das pontuações obtidas por crianças em um teste de precisão de leitura. Para cada conjunto de dados, a função de log-verossimilhança foi maximizada numerica-

mente através dos métodos de otimização não-linear estudados nos Capítulos 3 e 4. Com a finalidade de avaliar a eficiência e a precisão de cada método, foram comparados os valores obtidos para as estimativas pontuais e seus erros-padrão assim como os valores máximos das funções de log-verossimilhança. Adicionalmente, foi observado o comportamento das estimativas dos parâmetros e, quando necessário, do gradiente da função em cada uma das iterações realizadas por cada um dos métodos.

5.2 Aplicação a Dados de Gasolina de Prater

A variável dependente é a proporção de petróleo cru transformado em gasolina após o processo de destilação e fracionamento. As covariáveis são: temperatura em que 10% do petróleo cru é vaporizado e temperatura (^{o}F) em que toda a gasolina evapora. Essas covariáveis correspondem aos diferentes tipos de petróleo cru submetidos a experimentação sob diferentes condições de destilação. O conjunto de dados contém 32 observações. A primeira covariável assume dez valores diferentes, que são usados para definir os dez níveis de petróleo cru. A especificação do modelo para a resposta média é

$$g(\mu_t) = \beta_0 + \beta_1 x_{t1} + \beta_2 x_{t2} + \cdots + \beta_9 x_{t9} + \beta_{10} x_{t10}, \quad t = 1, \ldots, 32, \tag{5.1}$$

em que $g(\cdot)$ é a função de ligação logit, $x_{t1}, \ldots, x_{t9}$ representam nove variáveis *dummy* para os primeiros nove níveis do petróleo cru e x_{t10} mede a temperatura em que a gasolina é vaporizada.

A Tabela 5.1 apresenta as estimativas dos parâmetros do modelo (5.1) e os respectivos erros-padrão assintóticos. Observa-se que as estimativas obtidas através dos métodos BFGS e Newton são iguais, sendo o valor máximo encontrado da função de verossimilhança de

84.7976. Adicionalmente, notamos que os métodos simplex e CG não convergiram em 1000 iterações. Porém, as estimativas dos parâmetros de regressão fornecidas pelo método CG na iteração 1000 ficaram próximas das fornecidas pelos métodos que apresentaram convergência, a do parâmetro de precisão tendo ficado consideravelmente afastada do valor fornecido pelos outros métodos. Isso sugere que a maior dificuldade do método está em encontrar a estimativa de máxima verossimilhança desse parâmetro, fazendo com que o critério de parada não seja satisfeito. Tal como observado na Seção 4.3, os métodos BFGS e Newton tipicamente são capazes de fornecer com boa precisão uma estimativa para o parâmetro de locação do modelo, apresentando erros-padrão menores do que os dos demais métodos. Os métodos simplex e CG foram avaliados considerando-se maior número de iterações e o primeiro só convergiu após 12000 iterações, alcançando um valor máximo da função de log-verossimilhança igual a 83.2066, com estimativas dos parâmetros de locação e erros-padrão muito próximos dos obtidos até a iteração número 1000; a estimativa do parâmetro de precisão mudou consideravelmente, contudo, ficando em 338.5943 com erro-padrão de 84.609. No que tange ao método CG, ele não convergiu nem mesmo com 150000 iterações, fornecendo, ao término desse total de iterações, máximo de 82.9085 e estimativas dos parâmetros de locação e precisão muito semelhantes àquelas obtidas até a iteração número 1000. Dado que o valor máximo da função de log-verossimilhança fornecido pelos métodos BFGS e Newton foi superior ao valor máximo encontrado pelo método simplex, podemos considerar que as estimativas obtidas através daqueles métodos são preferíveis.

Cabe destacar a proximidade dos valores máximos da função de log-verossimilhança e das estimativas de ϕ pelos métodos BFGS, Newton e simplex, quando este último convergiu.

Parâmetro	**BFGS**	**Newton**	**Simplex****	**CG****
β_0	-6.1596	-6.1596	-6.4164	-6.1386
	(0.1823)	(0.1823)	(0.2277)	(0.2365)
β_1	1.7277	1.7277	1.7863	1.7227
	(0.1012)	(0.1012)	(0.1258)	(0.1314)
β_2	1.3226	1.3226	1.4043	1.3182
	(0.1179)	(0.1179)	(0.1463)	(0.1530)
β_3	1.5723	1.5723	1.6557	1.5668
	(0.1161)	(0.1161)	(0.1442)	(0.1507)
β_4	1.0597	1.0597	1.1063	1.0565
	(0.1024)	(0.1024)	(0.1271)	(0.1328)
β_5	1.1338	1.1338	1.1716	1.1299
	(0.1035)	(0.1035)	(0.1286)	(0.1344)
β_6	1.0402	1.0402	1.0796	1.0369
	(0.1060)	(0.1060)	(0.1317)	(0.1376)
β_7	0.5437	0.5437	0.5970	0.5426
	(0.1091)	(0.1091)	(0.1352)	(0.1415)
β_8	0.4959	0.4959	0.5394	0.4945
	(0.1089)	(0.1089)	(0.1350)	(0.1413)
β_9	0.3858	0.3858	0.4148	0.3847
	(0.1186)	(0.1186)	(0.1469)	(0.1539)
β_{10}	0.0110	0.0110	0.0115	0.0109
	(0.0004)	(0.0004)	(0.0005)	(0.0005)
ϕ	440.2784	440.2784	288.8015	259.5773
	(110.0256)	(110.0256)	(72.16055)	(64.85121)
Máximo	84.7976	84.7976	82.7459	82.9088

** *Indica não-convergência em 1000 iterações.*

Tabela 5.1: Estimativas obtidas para o conjunto de dados de gasolina de Prater.

Neste conjunto de dados foi detectado um ponto de alavanca, o qual foi removido, tendo sido os parâmetros do modelo então estimados novamente. Os resultados obtidos encontram-se apresentados na Tabela 5.2, onde é possível observar que o maior impacto do ponto de alavanca reside na estimação do parâmetro ϕ, cuja estimativa aumentou notavelmente (mudança relativa de 31.23%) para os métodos BFGS e Newton, que foram os únicos que alcançaram convergência. Adicionalmente, pode-se observar que os erros-padrão das esti-

mativas dos parâmetros de locação não apresentaram mudanças significativas; entretanto, o erro-padrão da estimativa do parâmetro de precisão aumentou.

Parâmetro	**BFGS**	**Newton**	**Simplex****	**CG****
β_0	-6.3565	-6.3565	-6.5777	-6.4951
	(0.1716)	(0.1716)	(0.2141)	(0.2307)
β_1	1.8869	1.8869	1.8944	1.9381
	(0.1002)	(0.1002)	(0.1233)	(0.1344)
β_2	1.3704	1.3704	1.3855	1.4232
	(0.1042)	(0.1042)	(0.1283)	(0.1397)
β_3	1.6251	1.6251	1.6558	1.6809
	(0.1028)	(0.1028)	(0.1264)	(0.1379)
β_4	1.0807	1.0807	1.0674	1.1104
	(0.0898)	(0.0898)	(0.1101)	(0.1205)
β_5	1.1516	1.1516	1.1247	1.1790
	(0.0907)	(0.0907)	(0.1113)	(0.1217)
β_6	1.0577	1.0577	1.0517	1.0842
	(0.0929)	(0.0929)	(0.1139)	(0.1247)
β_7	0.5652	0.5652	0.5193	0.5985
	(0.0956)	(0.0956)	(0.1178)	(0.1281)
β_8	0.5007	0.5007	0.4654	0.5182
	(0.0953)	(0.0953)	(0.1170)	(0.1279)
β_9	0.3852	0.3852	0.3450	0.3983
	(0.1038)	(0.1038)	(0.1274)	(0.1392)
β_{10}	0.0115	0.0115	0.0121	0.0118
	(0.0004)	(0.0004)	(0.0005)	(0.0005)
ϕ	577.7907	577.7907	379.3334	322.2157
	(146.7204)	(146.7204)	(96.31674)	(81.80703)
Máximo	86.6187	86.6187	84.1378	84.1647

** *Indica não-convergência em 1000 iterações.*

Tabela 5.2: Estimativas obtidas para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

Notamos que o método simplex agora apresentou convergência após 7400 iterações, o que equivale aproximadamente à vigésima parte das iterações efetuadas até convergência na presença do ponto de alavanca. Para este método, as estimativas dos parâmetros de locação são muito próximas das fornecidas pelos métodos que tiveram rápida convergência, enquanto

a estimativa obtida para o parâmetro de precisão foi de 412.9487 com valor máximo da função de verossimilhança igual a 84.6132. Este valor é bem menor do que o alcançado pelos métodos BFGS e Newton, indicando que o ponto atingido não é aquele que maximiza a função.

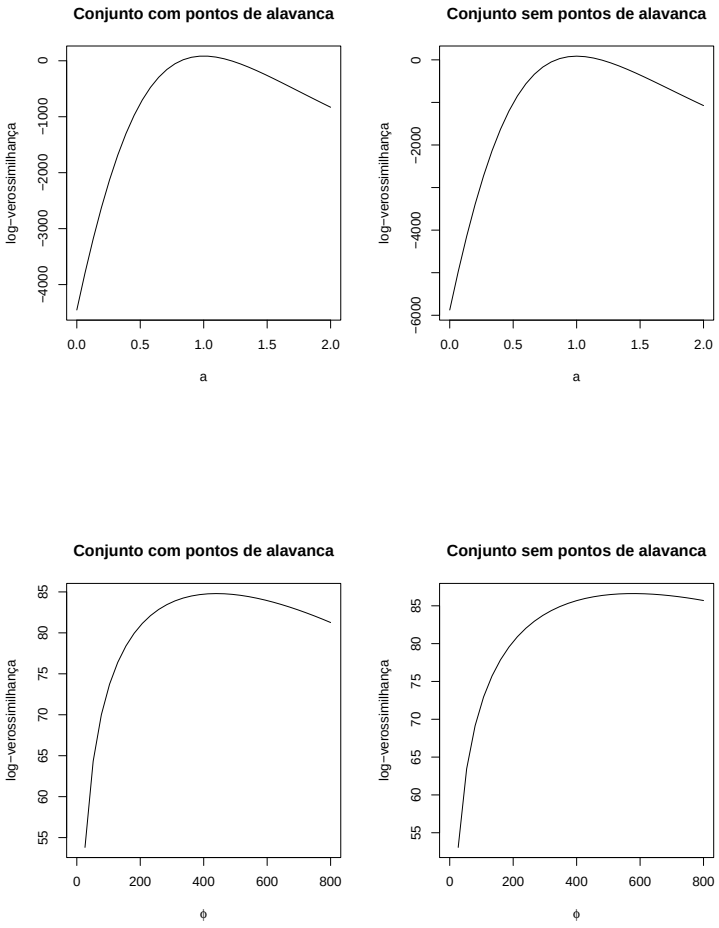

Figura 5.1: Funções de log-verossimilhança para diferentes valores dos parâmetros de locação e precisão no conjunto de dados de gasolina de Prater.

Dada a diferença notável entre as estimativas obtidas através dos métodos BFGS, Newton e simplex para o parâmetro ϕ, foram construídos gráficos com o objetivo de examinar como variações em $\beta = \{\beta_0, \beta_1, \ldots, \beta_{10}\}$ e em ϕ alteram o valor da função de log-verossimihança $\ell(\beta, \phi)$. Em primeiro lugar, foi investigado o impacto de variações no vetor de parâmetros de locação β; este vetor, todavia, possui mais de dois parâmetros e não é possível fazer o gráfico em três dimensões. Portanto, foi fixado o valor de ϕ como aquele estimado pelo método BFGS (e de Newton) e o valor de β foi variado em forma linear, com a função $\ell(\beta, \phi)$ substituída por $\ell(a\widehat{\beta}, \phi)$. Os valores estabelecidos de $\widehat{\beta}$ foram aqueles encontrados pelo método BFGS (e de Newton); o valor da constante a foi variado. Posteriormente, foi observada a variação da função de log-verossimilhança para diferentes valores do parâmetro de precisão ϕ, tendo sido considerados os valores $\widehat{\beta}$ obtidos através dos métodos BFGS e Newton como fixos e o parâmetro ϕ como variável independente. Este procedimento foi realizado tanto para os dados com ponto de alavanca quanto para o conjunto de dados sem este ponto.

Os gráficos de avaliação são apresentados na Figura 5.1. No que se refere à variação do vetor β, pode-se notar a concavidade da curva que evidencia que o ponto de máximo está no valor $a = 1$, ou seja, a otimização pelos métodos BFGS e Newton foi bem sucedida. No que tange ao parâmetro de precisão, observa-se que a concavidade da função se vê significativamente afetada pela eliminação do ponto de alavanca. Verifica-se que, no conjunto de dados que inclui o ponto de alavanca, há pouca variação na função para valores de ϕ entre 200 e 500, aproximadamente. No caso em que não há ponto de alavanca, o intervalo em que a variação de ϕ produz pouca variação na função se estende aproximadamente de 400 a 700.

Assim, pode-se concluir que a dificuldade dos métodos para fornecer estimativas precisas de ϕ deve-se ao fato de que uma variação no parâmetro ϕ não implica necessariamente uma variação significativa na função de log-verossimilhança.

As Figuras 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8 e 5.9 mostram o comportamento das estimativas e gradientes de cada parâmetro do modelo nos métodos BFGS e Newton, tanto para o conjunto de dados com o ponto de alavanca quanto para o conjunto sem esse dado. Pode-se observar que, tal como ocorrido na simulação, o método de Newton se mostrou mais eficiente em termos do número de iterações até convergência e mais robusto à presença do ponto de alavanca, pois não houve variação significativa no comportamento das iterações quando este ponto foi retirado dos dados. Nota-se também a dificuldade do método BFGS para obter a estimativa do parâmetro de precisão, pois ele ficou preso em um mesmo valor em várias iterações.

Analogamente, as Figuras 5.10, 5.11, 5.12 e 5.13 mostram o comportamento das estimativas dos parâmetros em cada uma das primeiras 1000 iterações efetuadas pelos métodos simplex e CG para os dados com e sem ponto de alavanca. Evidencia-se a alta sensibilidade do método simplex à presença do ponto de alavanca, pois o comportamento foi claramente afetado por tal ponto. Por sua parte, o método CG se mostrou mais estável, com pouca variação no comportamento das estimativas ao longo das iterações, após a retirada do ponto de alavanca.

Finalmente, pode-se concluir que para este conjunto de dados o método mais eficiente na estimação dos parâmetros de locação e precisão do modelo (5.1) foi o método de Newton, por sua robustez à presença do ponto de alavanca e pela rapidez na obtenção das estimativas.

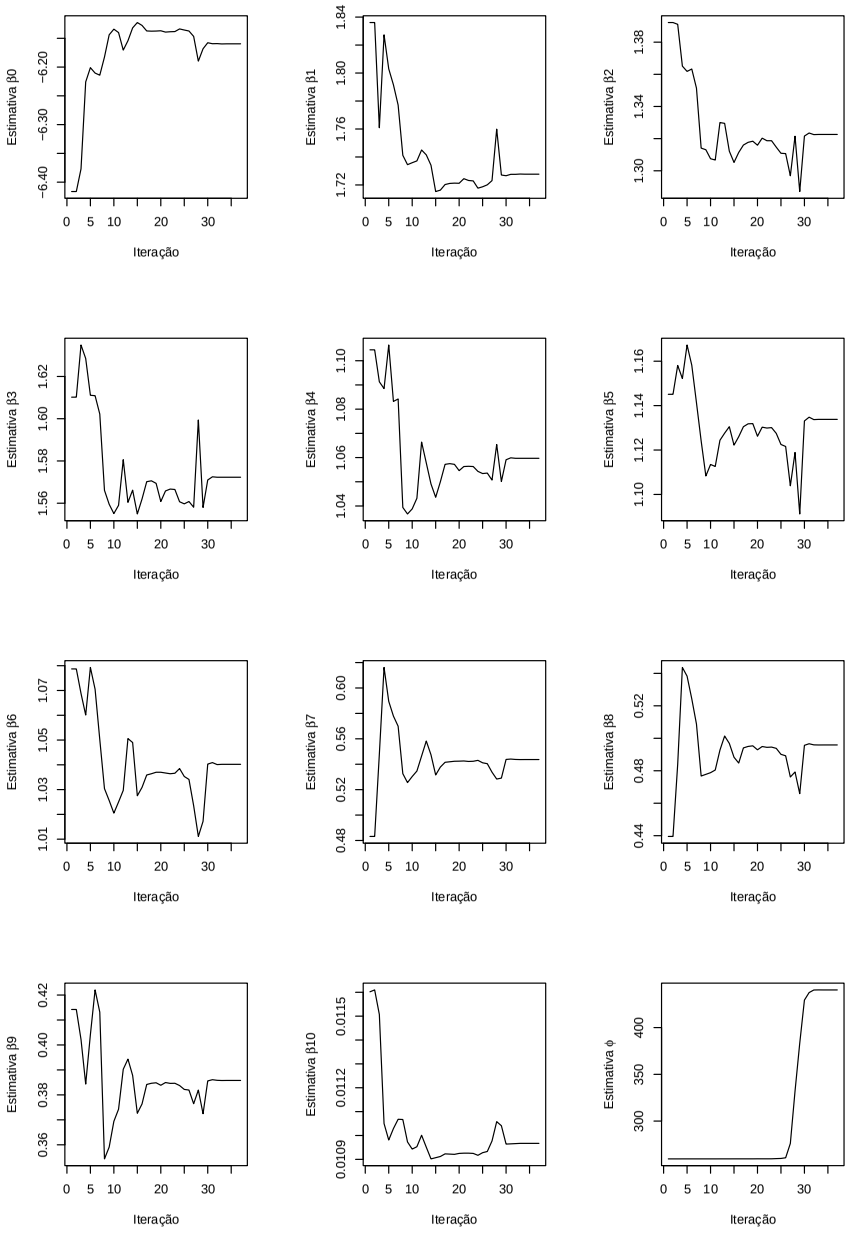

Figura 5.2: Comportamento das estimativas dos parâmetros em cada iteração do método BFGS para o conjunto de dados de gasolina de Prater.

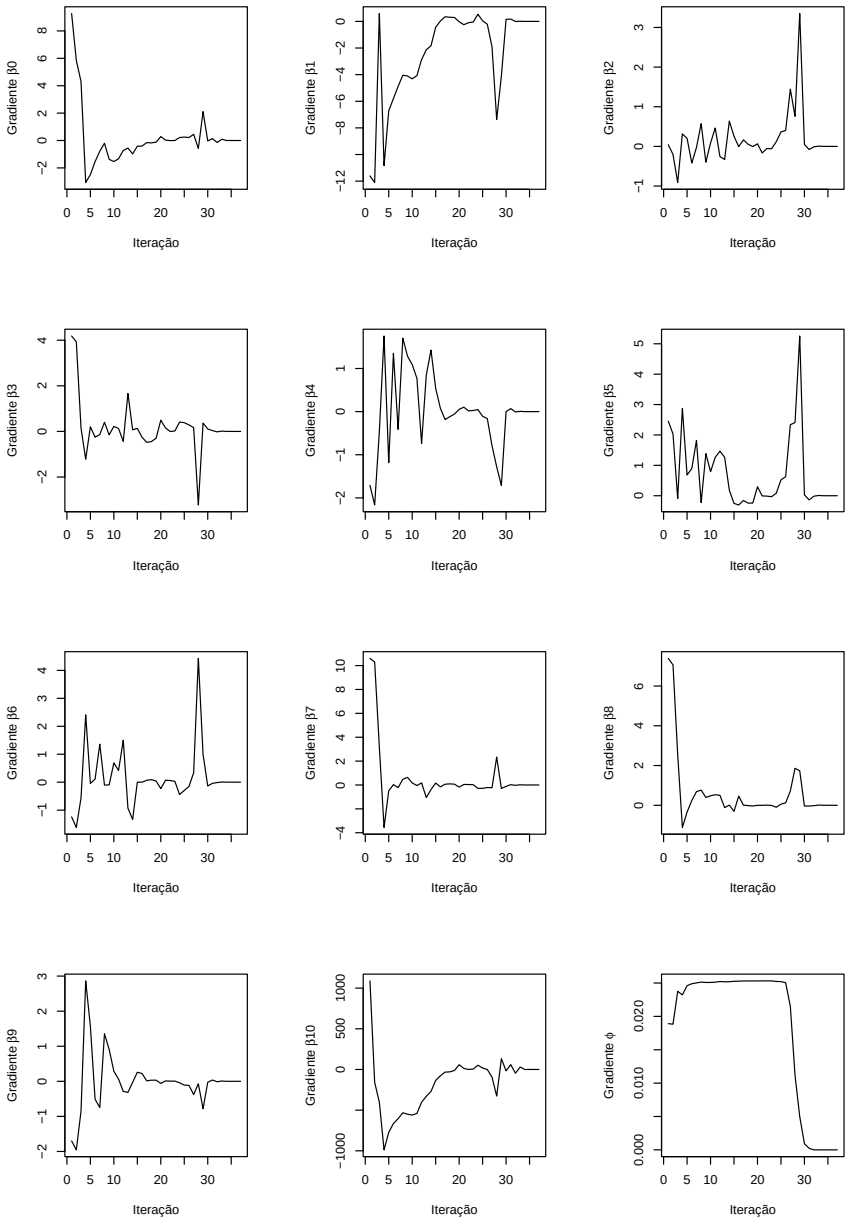

Figura 5.3: Comportamento dos gradientes em cada iteração do método BFGS para o conjunto de dados de gasolina de Prater.

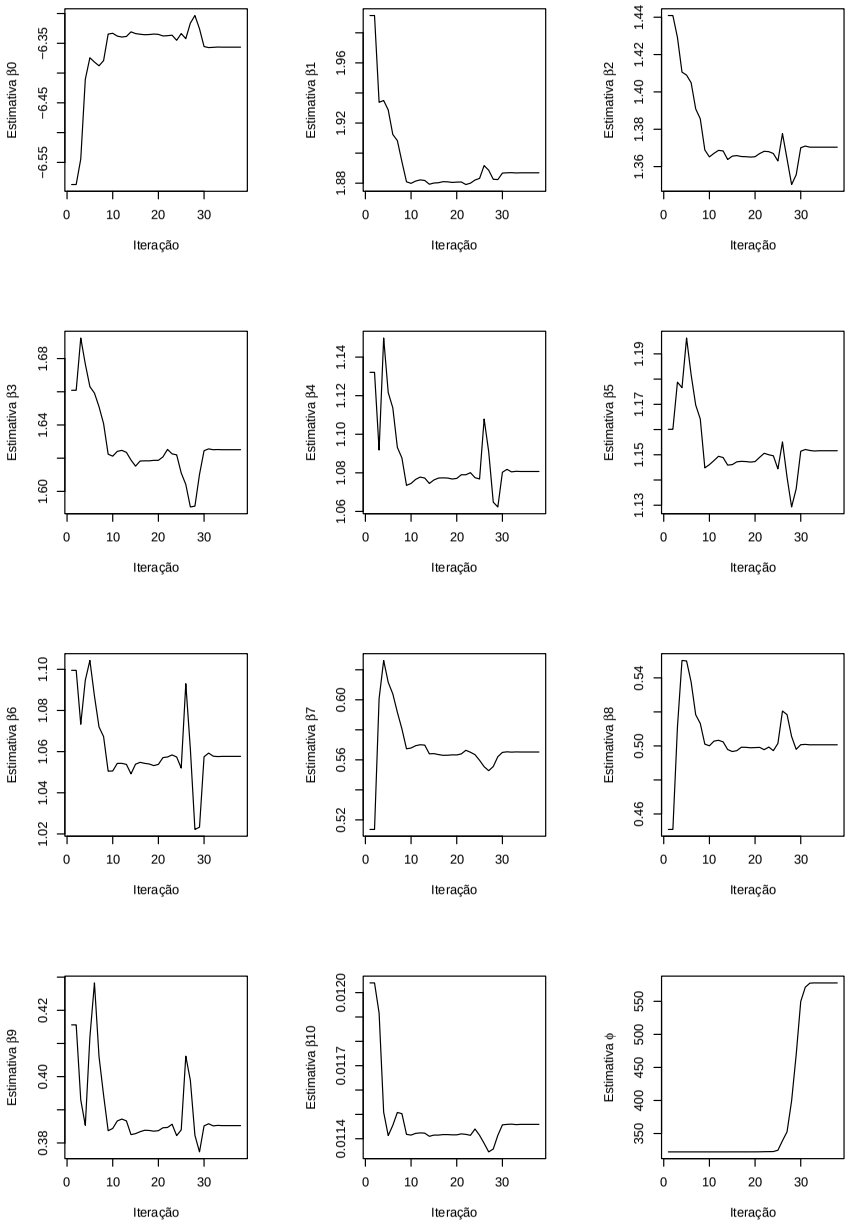

Figura 5.4: Comportamento das estimativas dos parâmetros em cada iteração do método BFGS para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

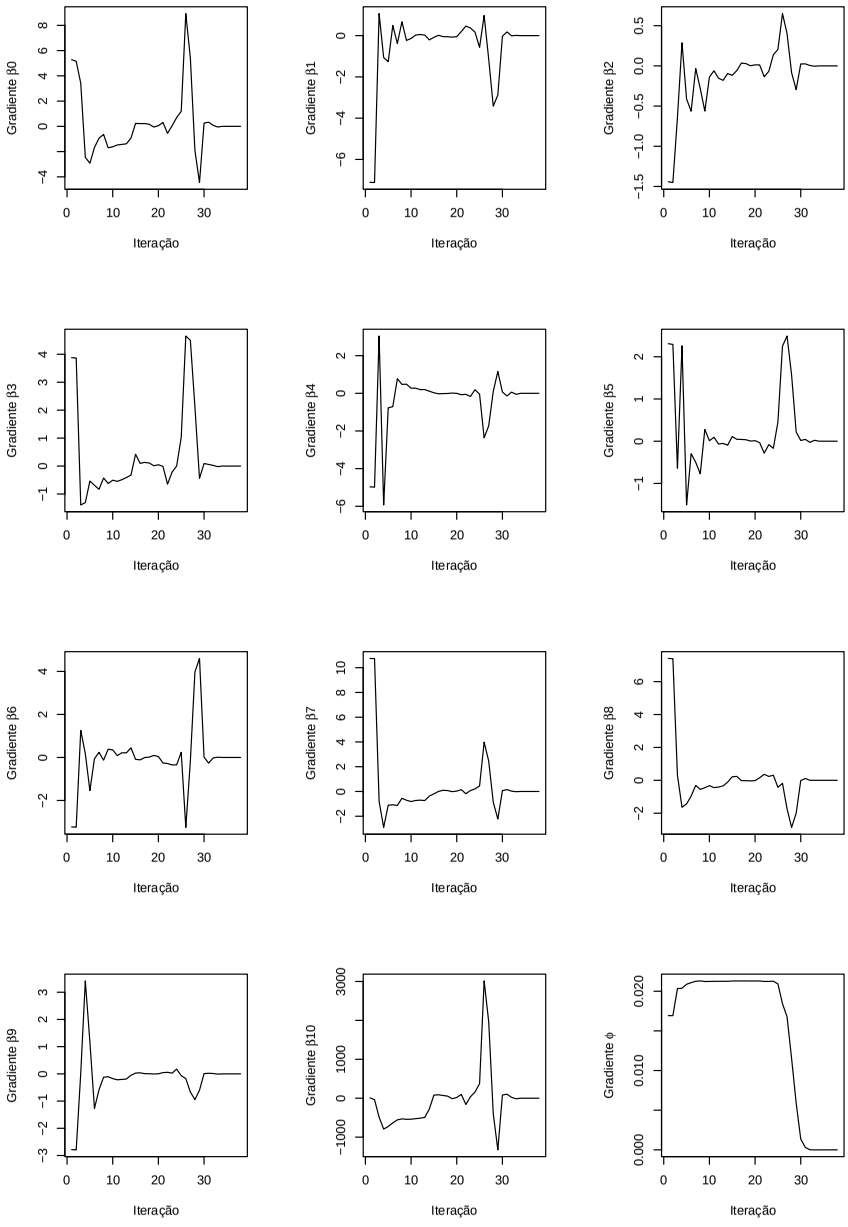

Figura 5.5: Comportamento dos gradientes em cada iteração do método BFGS para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

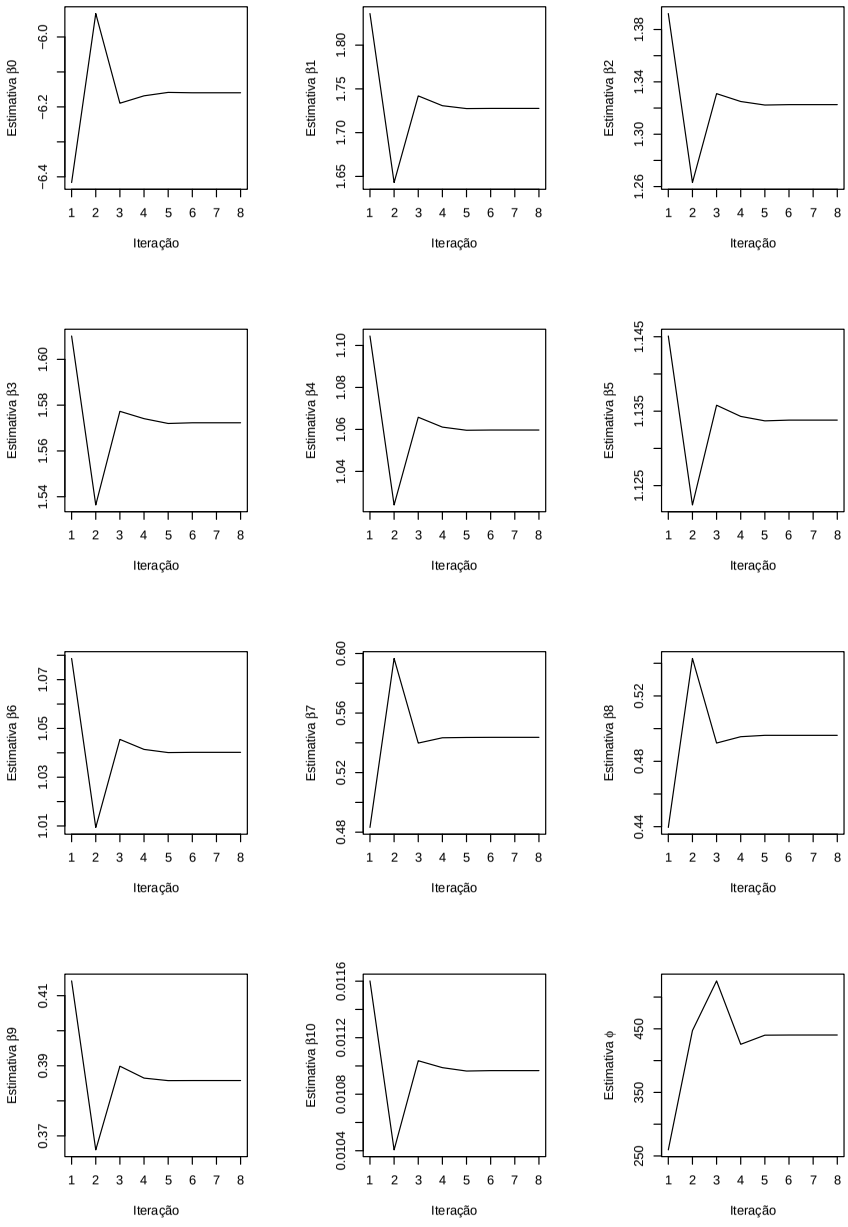

Figura 5.6: Comportamento das estimativas dos parâmetros em cada iteração do método de Newton para o conjunto de dados de gasolina de Prater.

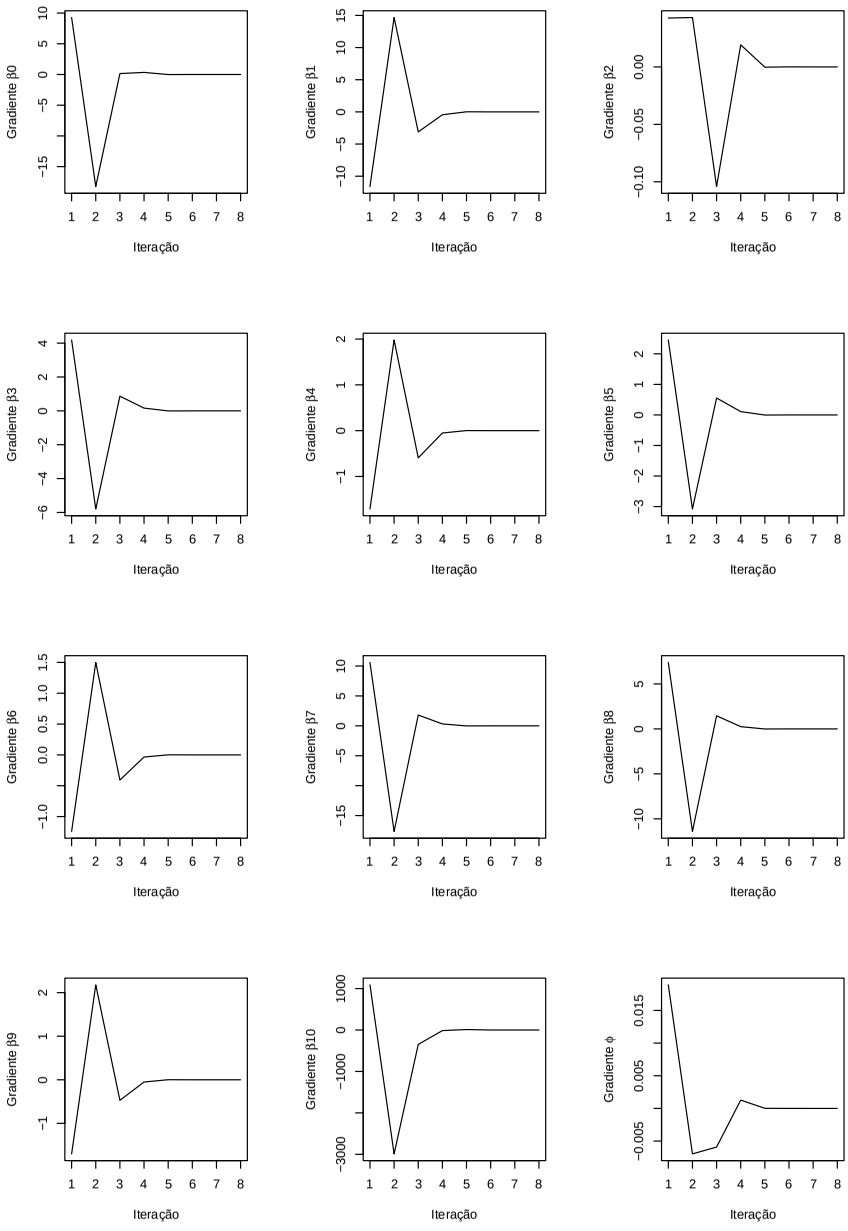

Figura 5.7: Comportamento dos gradientes em cada iteração do método de Newton para o conjunto de dados de gasolina de Prater.

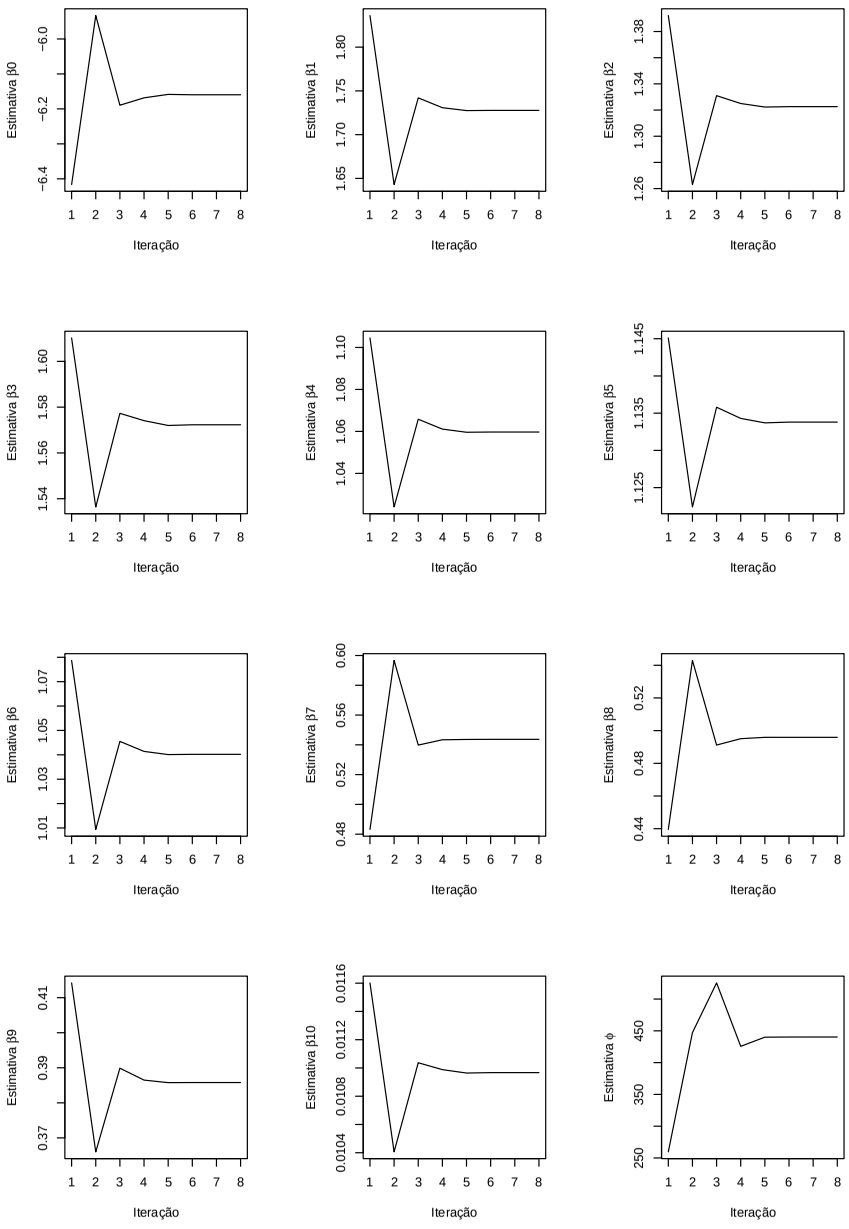

Figura 5.8: Comportamento das estimativas dos parâmetros em cada iteração do método de Newton para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

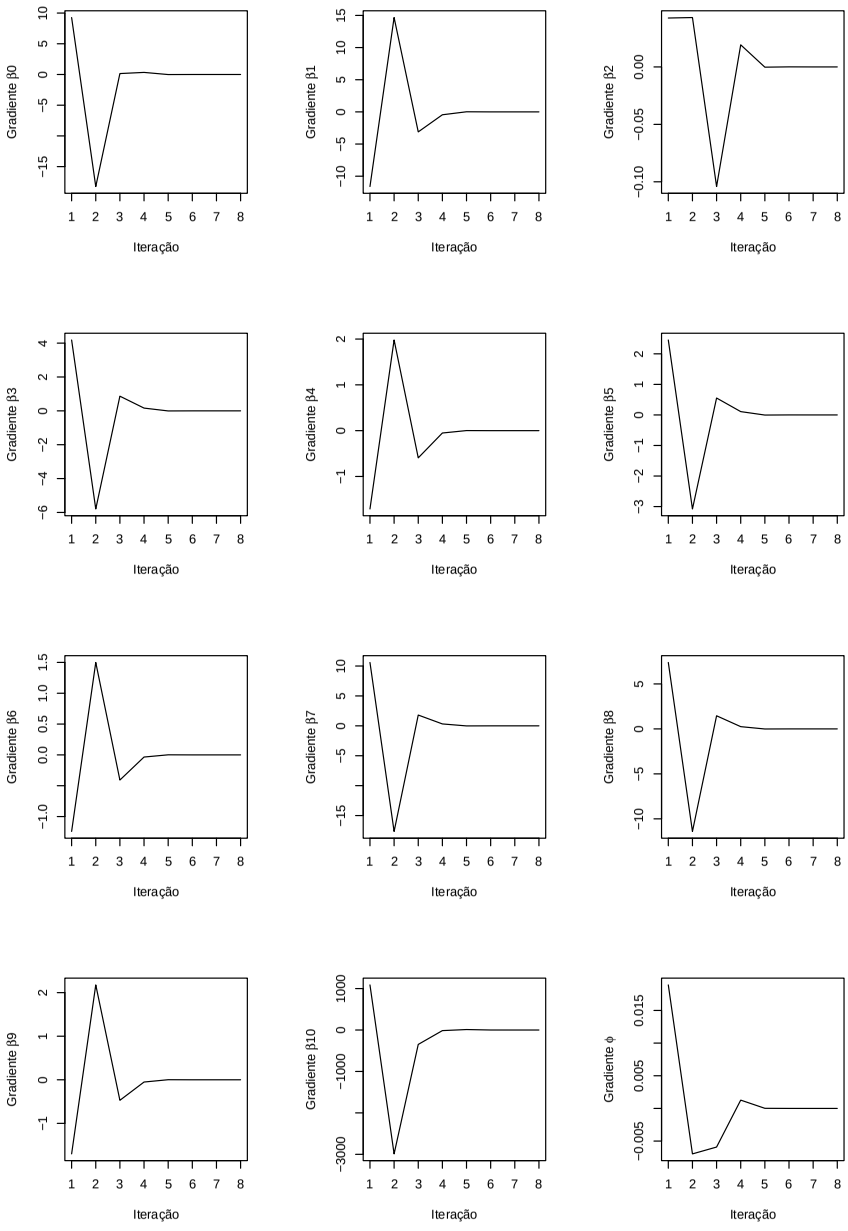

Figura 5.9: Comportamento dos gradientes em cada iteração do método de Newton para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

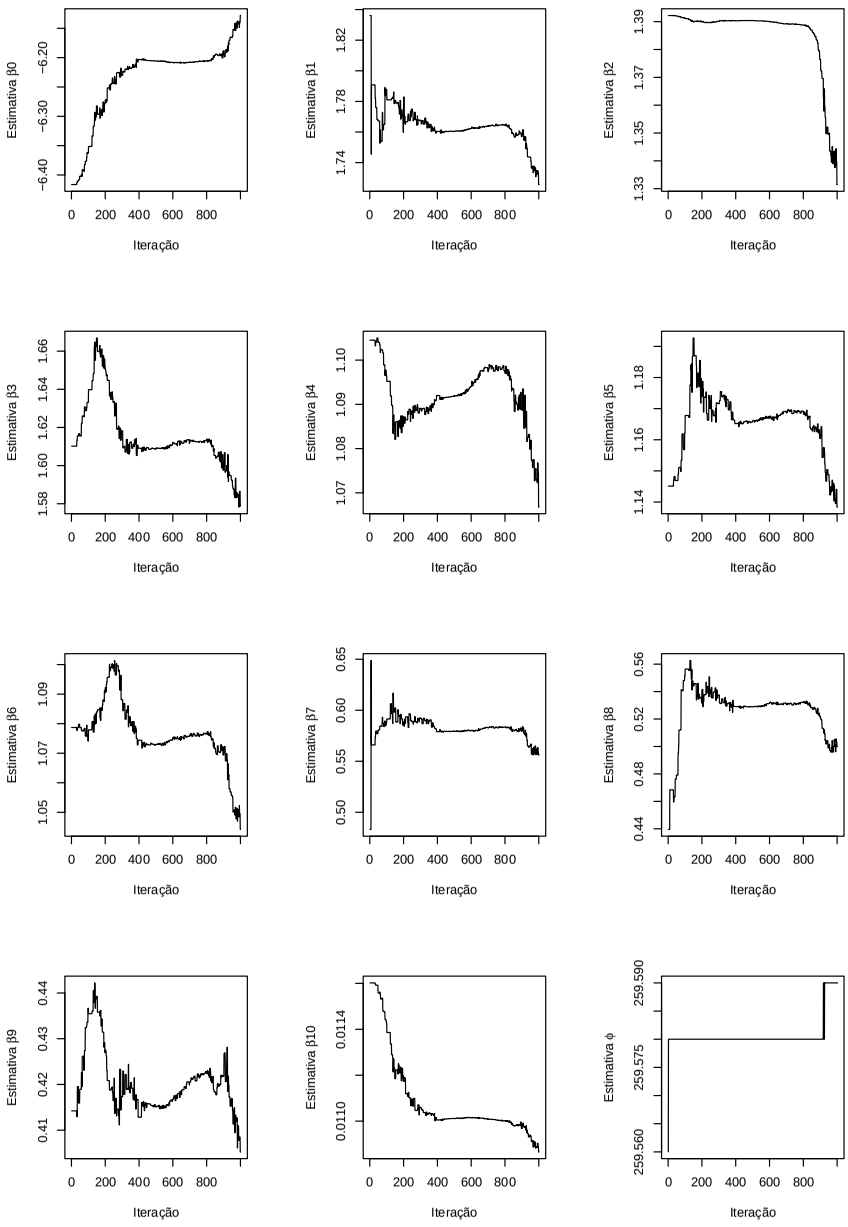

Figura 5.10: Comportamento das estimativas dos parâmetros em cada iteração do método simplex para o conjunto de dados de gasolina de Prater.

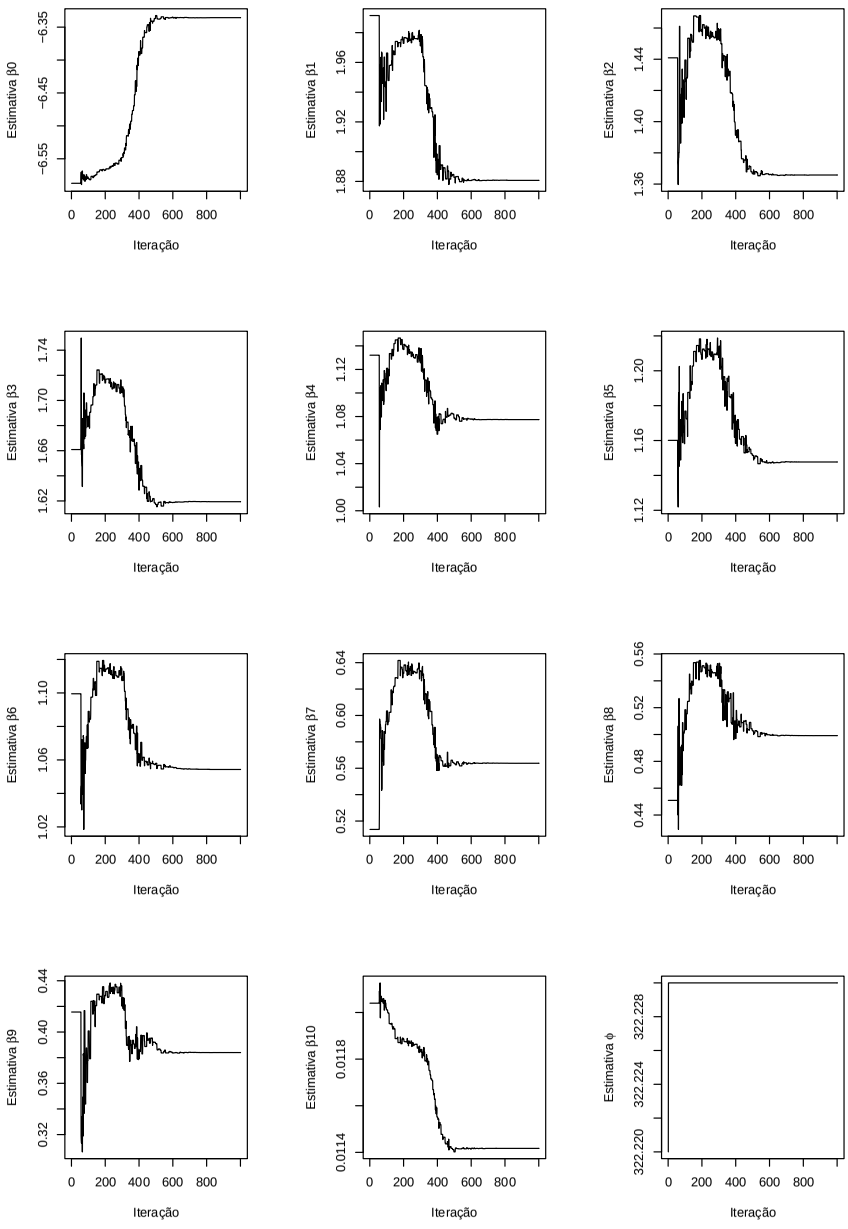

Figura 5.11: Comportamento das estimativas dos parâmetros em cada iteração do método simplex para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

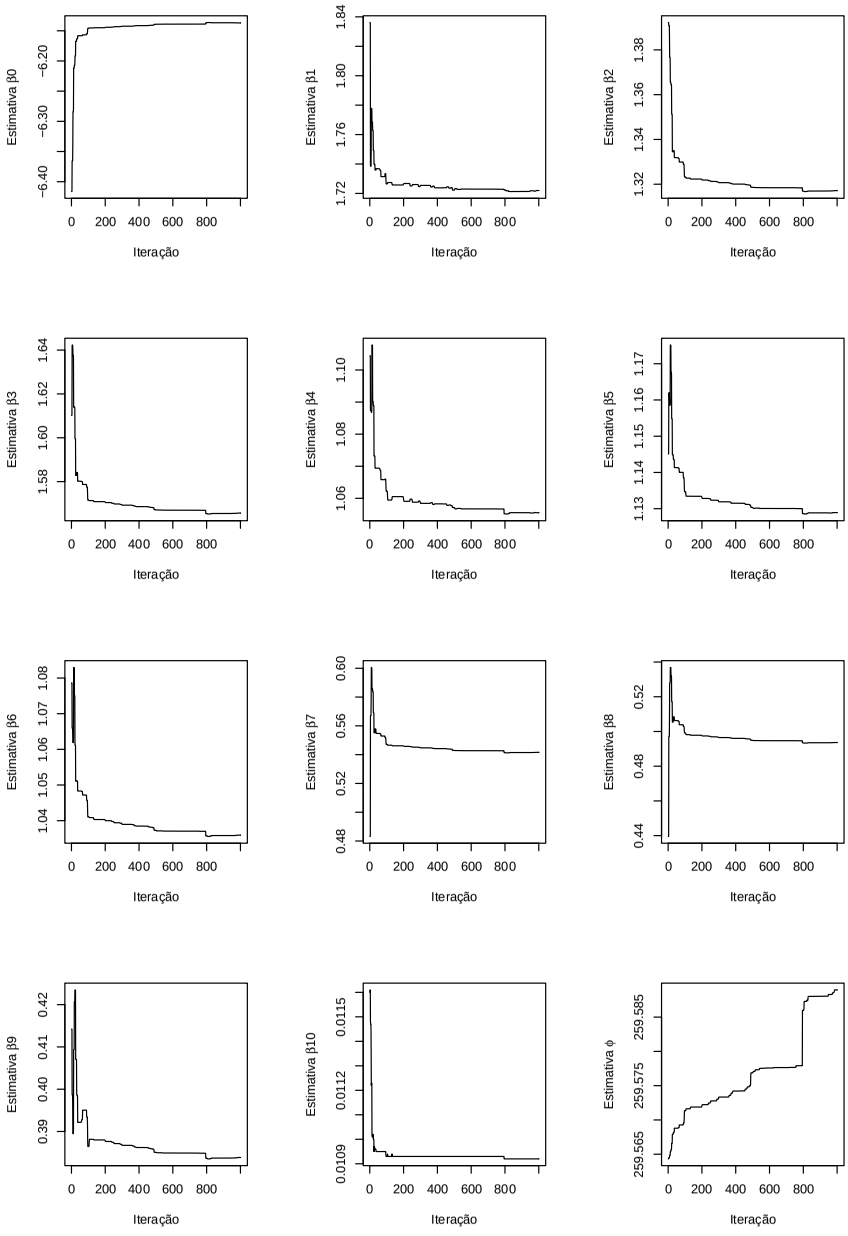

Figura 5.12: Comportamento das estimativas dos parâmetros em cada iteração do método gradiente conjugado para o conjunto de dados de gasolina de Prater.

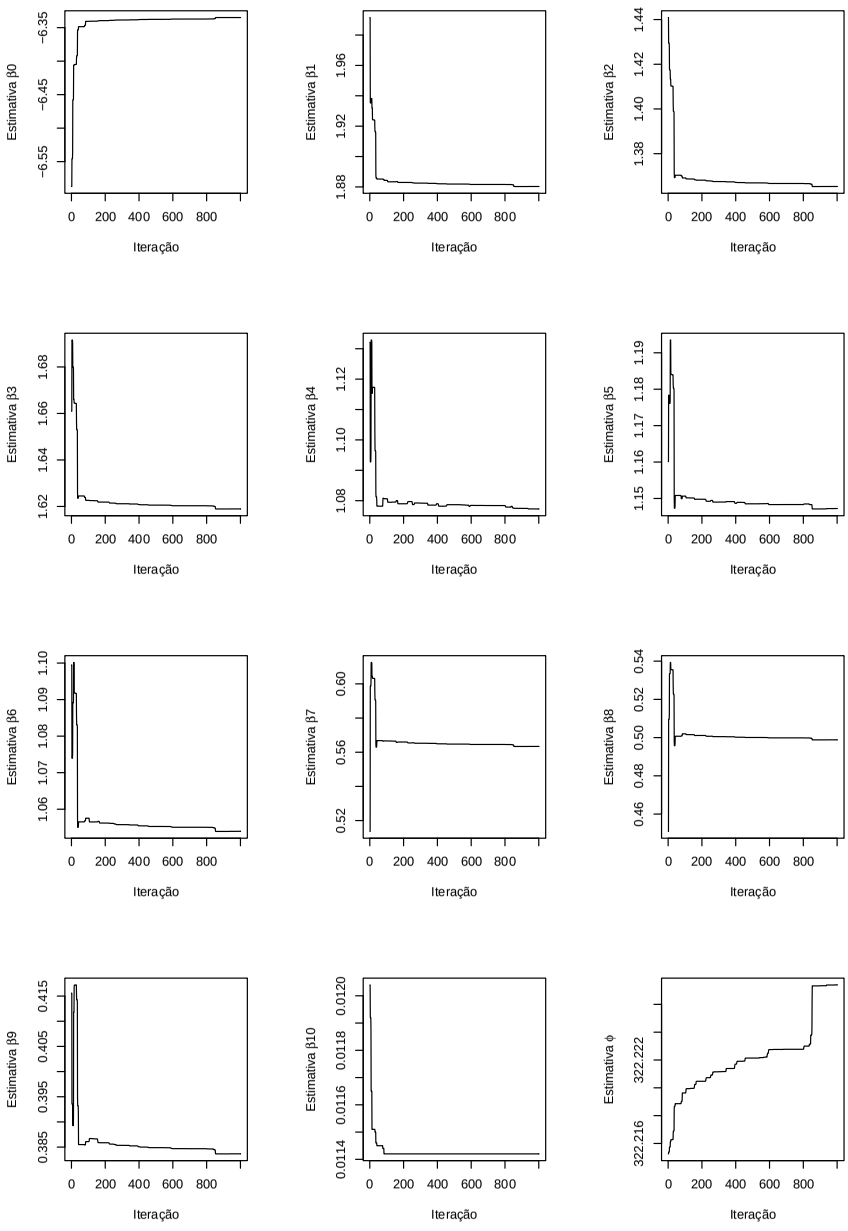

Figura 5.13: Comportamento das estimativas dos parâmetros em cada iteração do método gradiente conjugado para o conjunto de dados de gasolina de Prater sem pontos de alavanca.

5.3 Aplicação a Dados de Quociente Intelectual (QI)

Esta aplicação utiliza os dados de Pammer & Kevan (2004), posteriormente analisados por Smithson & Verkuilen (2006). A variável resposta são pontuações obtidas por 44 crianças em um teste de leitura e as covariáveis são: condição de dislexia ou não-dislexia (x_1), quociente intelectual não-verbal convertido a pontuações z (x_2) e uma variável de interação (x_3). Participaram do estudo 19 crianças disléxicas e 25 controles com idades entre 8 anos e 5 meses e 12 anos e 3 meses, que foram selecionadas de escolas primárias na cidade de Canberra, capital da Austrália. A covariável x_1 assume valor 1 quando a criança é disléxica e -1 caso contrário. As pontuações observadas foram transformadas linearmente de sua escala original ao intervalo unitário aberto $(0, 1)$, tomando primeiro $y' = (y-a)/(b-a)$, onde a e b representam a maior e a menor pontuação possível, respectivamente; posteriormente, a escala foi comprimida para evitar zeros e uns tomando $y'' = [y'(n-1) + 0.5]/n$, onde n representa o tamanho amostral.

Para este conjunto de dados, a estrutura do modelo de regressão beta usada é

$$g(\mu_t) = \beta_0 + \beta_1 x_{t1} + \beta_2 x_{t2} + \beta_3 x_{t3}, \quad t = 1, \ldots, 44, \tag{5.2}$$

onde $g(\cdot)$ é a função de ligação logit. As estimativas e os respectivos erros-padrão dos parâmetros do modelo (5.2) encontram-se apresentados na Tabela 5.3. Todos os métodos apresentaram convergência, sendo o método de Newton o mais eficiente, com apenas 4 iterações realizadas, e o método simplex o mais lento, com 287 iterações. O método BFGS executou 10 iterações, enquanto que o método CG necessitou de 32 iterações para atingir o ponto ótimo.

Observa-se que o valor máximo da função de log-verossimilhança foi de 51.3504 para todos os métodos; de fato, tanto as estimativas pontuais quanto os erros-padrão obtidos foram iguais. É possível observar que o erro-padrão da estimativa do parâmetro de precisão é notavelmente maior do que os erros-padrão das estimativas dos parâmetros do preditor linear, os quais são muito semelhantes entre si.

Parâmetro	**BFGS**	**Newton**	**Simplex**	**CG**
β_0	1.3338	1.3338	1.3338	1.3338
	(0.1357)	(0.1357)	(0.1357)	(0.1357)
β_1	-0.9736	-0.9736	-0.9736	-0.9736
	(0.1335)	(0.1335)	(0.1335)	(0.1335)
β_2	0.1608	0.1608	0.1608	0.1608
	(0.1344)	(0.1344)	(0.1344)	(0.1344)
β_3	-0.2186	-0.2186	-0.2186	-0.2186
	(0.1345)	(0.1345)	(0.1345)	(0.1345)
ϕ	11.1332	11.1332	11.1332	11.1332
	(2.4435)	(2.4435)	(2.4435)	(2.4435)
Máximo	51.3504	51.3504	51.3504	51.3504

Tabela 5.3: Estimativas obtidas para o conjunto de dados de QI.

A análise dos resíduos do modelo revela que há um ponto de alavanca nos dados. Ele foi retirado e os parâmetros do modelo foram re-estimados. Verificou-se que o maior impacto exercido por esse ponto recai sobre as estimativas dos parâmetros β_2 e β_3, que apresentaram mudanças relativas de 65.6% e 48.5%, respectivamente. Como pode ser observado na Tabela 5.4, o impacto do ponto de alavanca na estimação do parâmetro ϕ é leve, sendo a mudança relativa na estimativa de apenas 7.7%. A estimativa menos afetada pela retirada deste ponto é a do intercepto do modelo (β_0), com mudança relativa de 6.0%. Pode-se observar também que o erro-padrão da estimativa do ϕ aumentou consideravelmente em comparação com o aumento da mesma medida para os parâmetros de locação do modelo.

Parâmetro	**BFGS**	**Newton**	**Simplex**	**CG**
β_0	1.2541	1.2541	1.2541	1.2541
	(0.1325)	(0.1325)	(0.1325)	(0.1325)
β_1	-0.8916	-0.8916	-0.8916	-0.8916
	(0.1306)	(0.1306)	(0.1306)	(0.1306)
β_2	0.2663	0.2663	0.2663	0.2663
	(0.1368)	(0.1368)	(0.13677)	(0.1368)
β_3	-0.3245	-0.3245	-0.3245	-0.3245
	(0.1369)	(0.13692)	(0.13692)	(0.1369)
ϕ	11.9920	11.9920	11.9920	11.9920
	(2.6549)	(2.6549)	(2.6549)	(2.6549)
Máximo	50.4688	50.4688	50.4688	50.4688

Tabela 5.4: Estimativas obtidas para o conjunto de dados de QI sem pontos de alavanca.

Para este conjunto de dados foi realizada a mesma análise gráfica efetuada na aplicação apresentada na Subseção 5.2. Da Figura 5.14, pode-se deduzir a razão pela qual todos os métodos atingiram o mesmo ponto: tanto variações no parâmetro de locação quanto variações no parâmetro de precisão conduzem a variações consideráveis na função de log-verossimilhança. Nota-se também que a concavidade da função de log-verossimilhança não apresentou mudança significativa após a eliminação do ponto de alavanca.

As Figuras 5.15, 5.16, 5.17, 5.18, 5.19, 5.20, 5.21 e 5.22 mostram o comportamento das estimativas dos parâmetros e gradientes em cada iteração dos métodos BFGS e Newton. Observa-se que, embora o comportamento do método de Newton em cada iteração tenha mudado devido à eliminação do ponto de alavanca, a velocidade de convergência do método não foi afetada, pois o número de iterações até convergência passou de 4 para 5, um aumento mínimo. O número de iterações realizadas pelo método BFGS aumentou de 10 para 12; nota-se que não houve mudança significativa no comportamento das estimativas e dos gradientes quando foi retirado o ponto de alavanca.

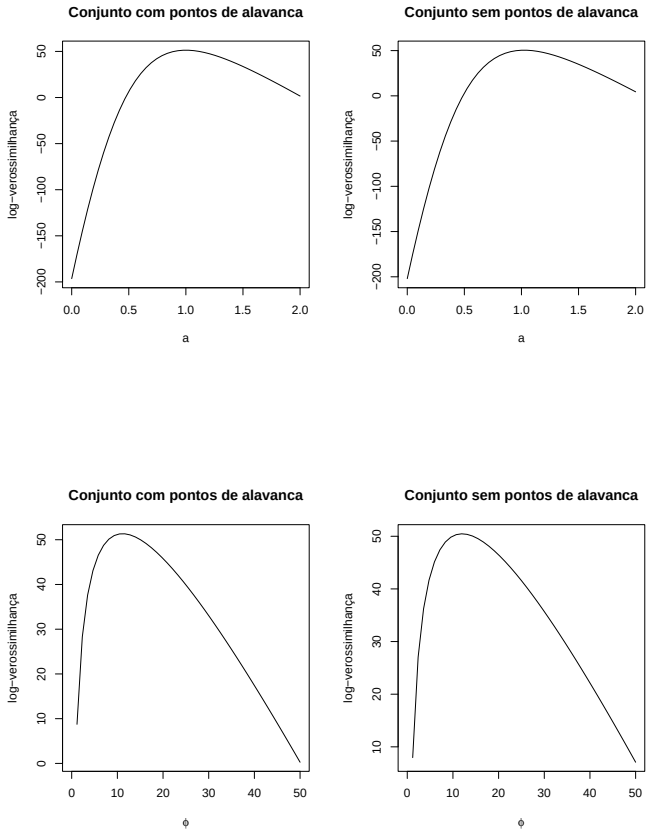

Figura 5.14: Funções de log-verossimilhança para diferentes valores dos parâmetros de locação e precisão no conjunto de dados de QI.

No que se refere aos métodos simplex e CG, os aumentos nos números de iterações até convergência após a retirada do ponto de alavanca foram de 11 e 95 iterações, respectivamente. As Figuras 5.23, 5.24, 5.25 e 5.26 evidenciam que quando foi removido o ponto de alavanca houve leve mudança no comportamento das estimativas dos parâmetros β_2 e β_3 obtidas através do método simplex, como era esperado devido ao grande impacto dessa observação sobre as estimativas destes parâmetros. Por sua parte, o método CG não mostrou mudança significativa de comportamento, porém o processo iterativo se tornou mais lento.

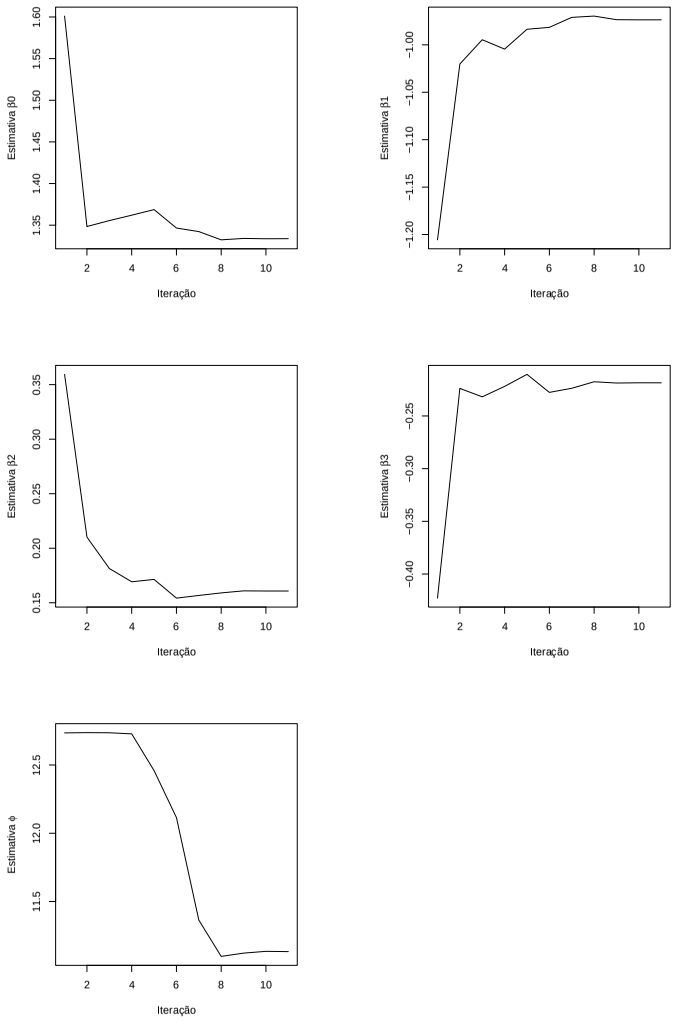

Figura 5.15: Comportamento das estimativas dos parâmetros em cada iteração do método BFGS para o conjunto de dados de QI.

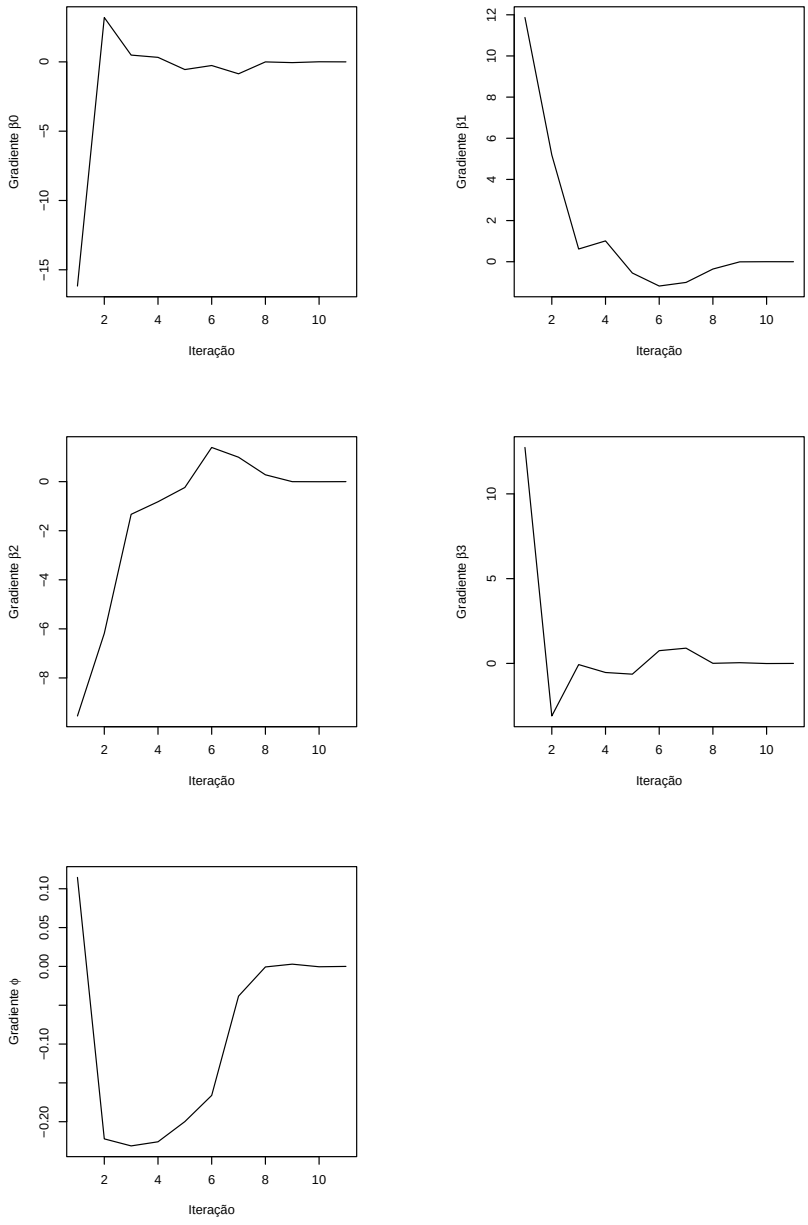

Figura 5.16: Comportamento dos gradientes em cada iteração do método BFGS para o conjunto de dados de QI com pontos de alavanca.

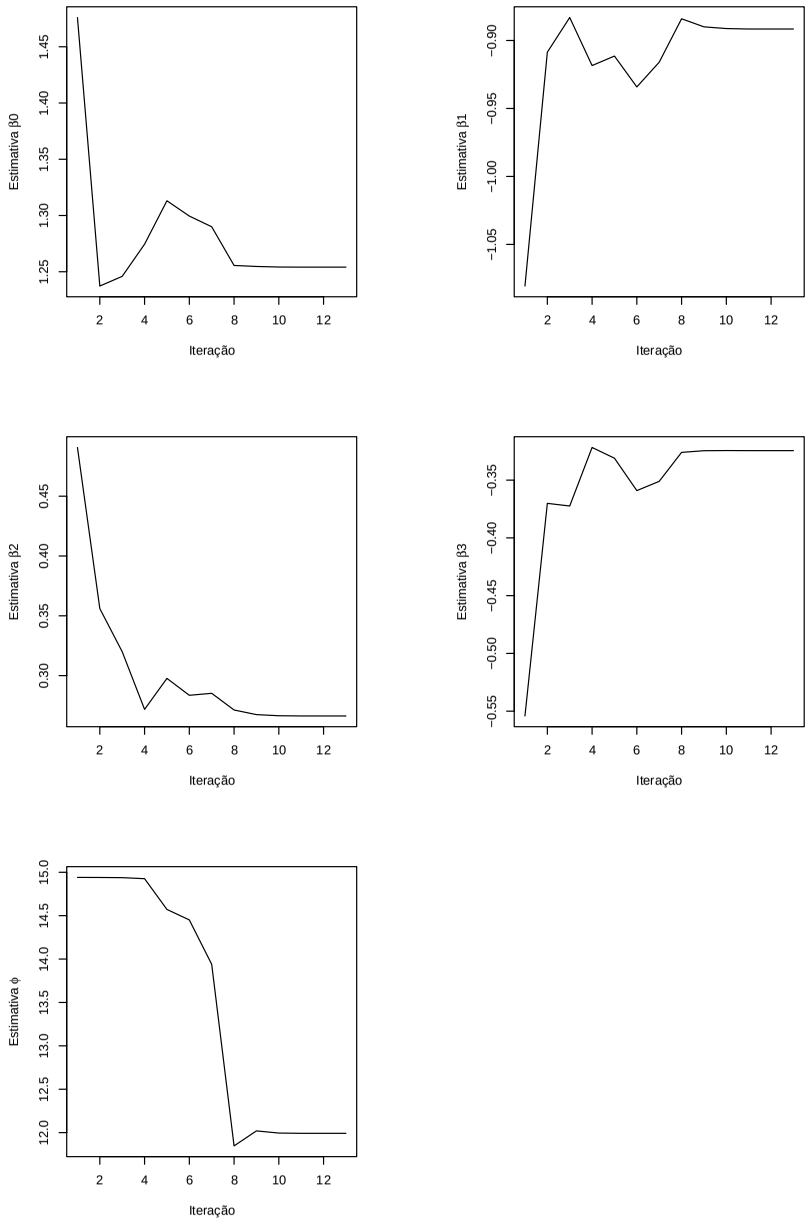

Figura 5.17: Comportamento das estimativas dos parâmetros em cada iteração do método BFGS para o conjunto de dados de QI.

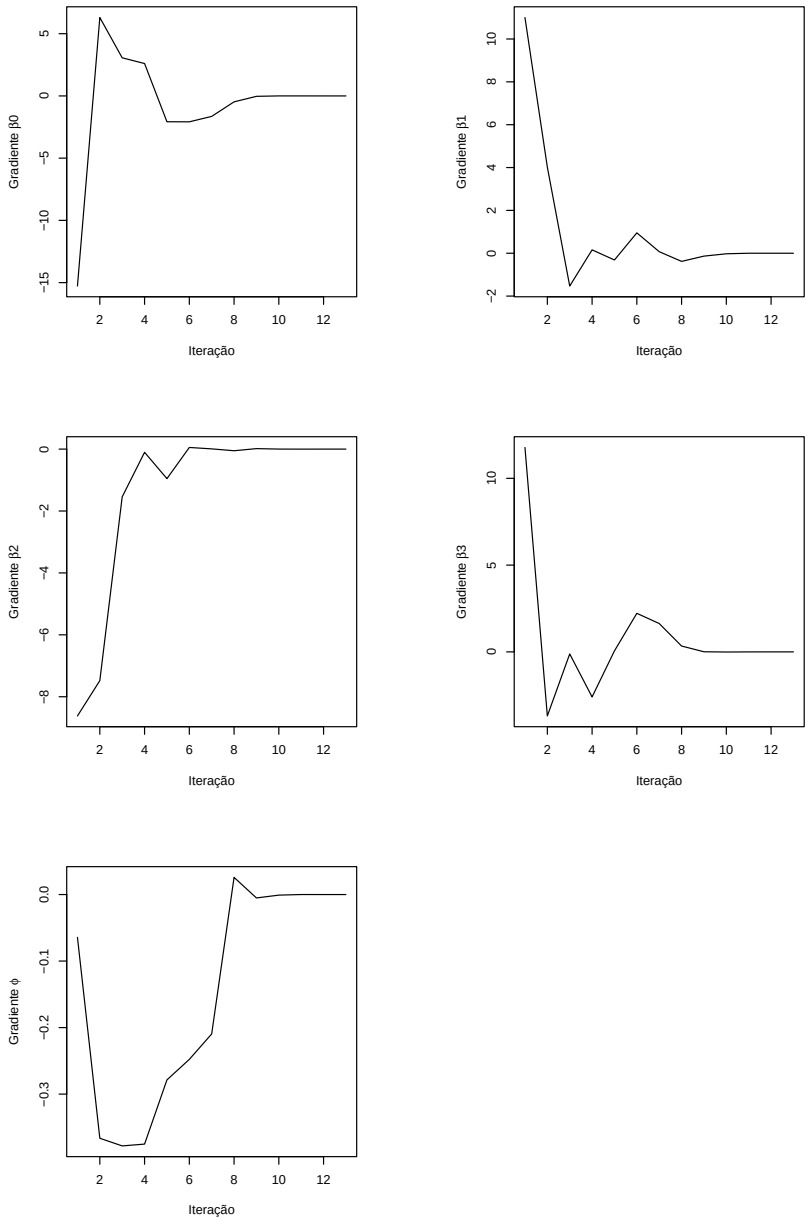

Figura 5.18: Comportamento dos gradientes em cada iteração do método BFGS para o conjunto de dados de QI sem pontos de alavanca.

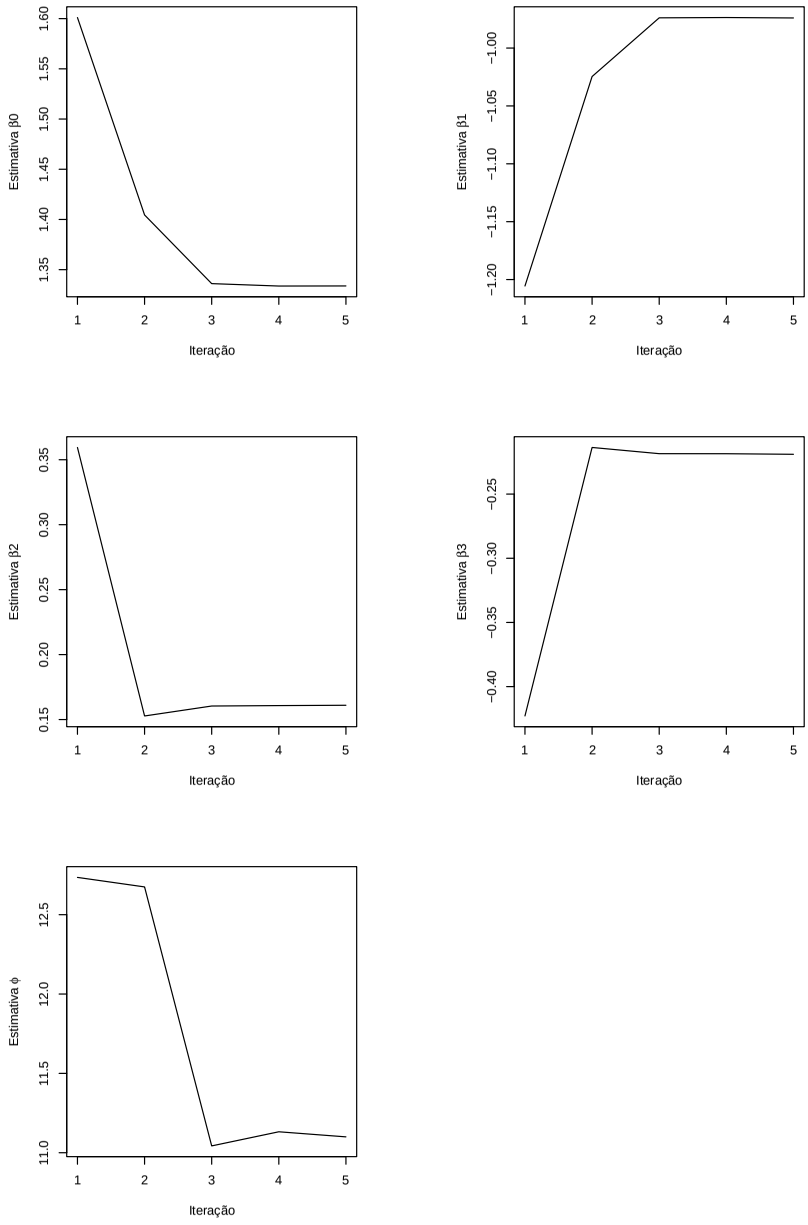

Figura 5.19: Comportamento das estimativas dos parâmetros em cada iteração do método de Newton para o conjunto de dados de QI.

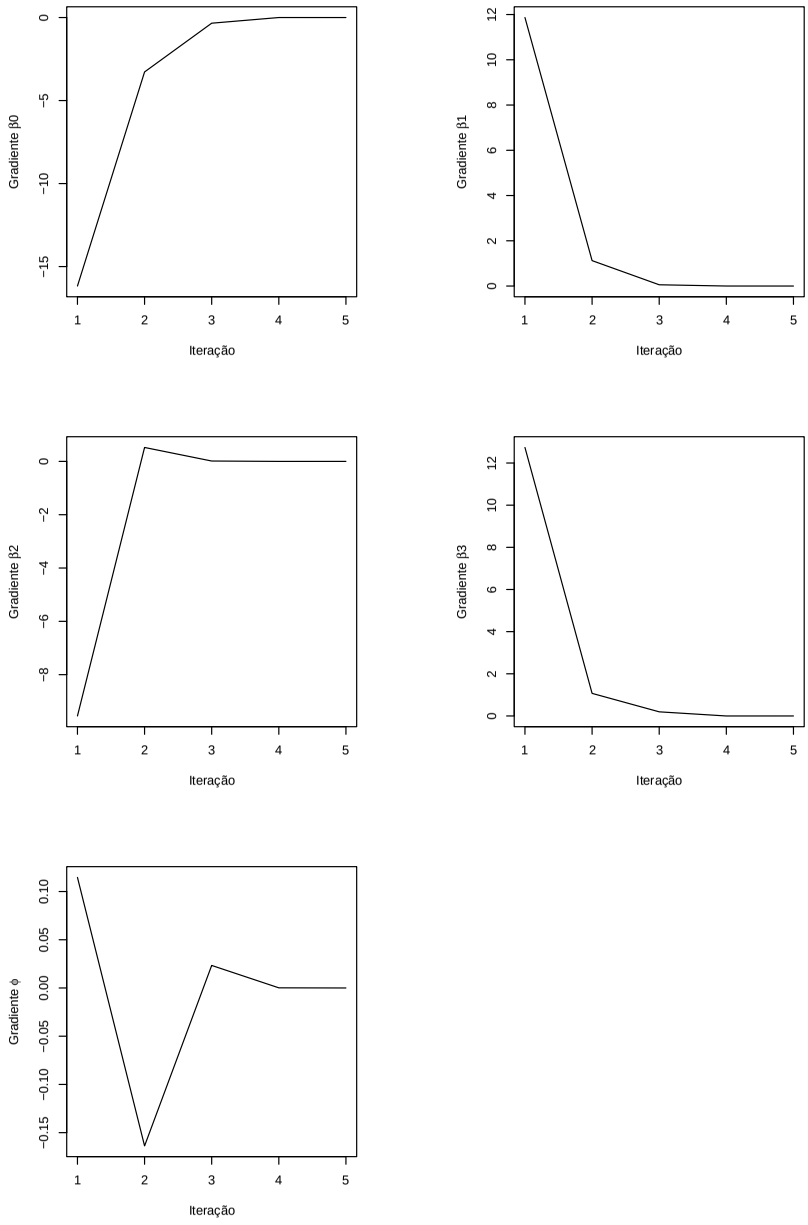

Figura 5.20: Comportamento dos gradientes em cada iteração do método de Newton para o conjunto de dados de QI.

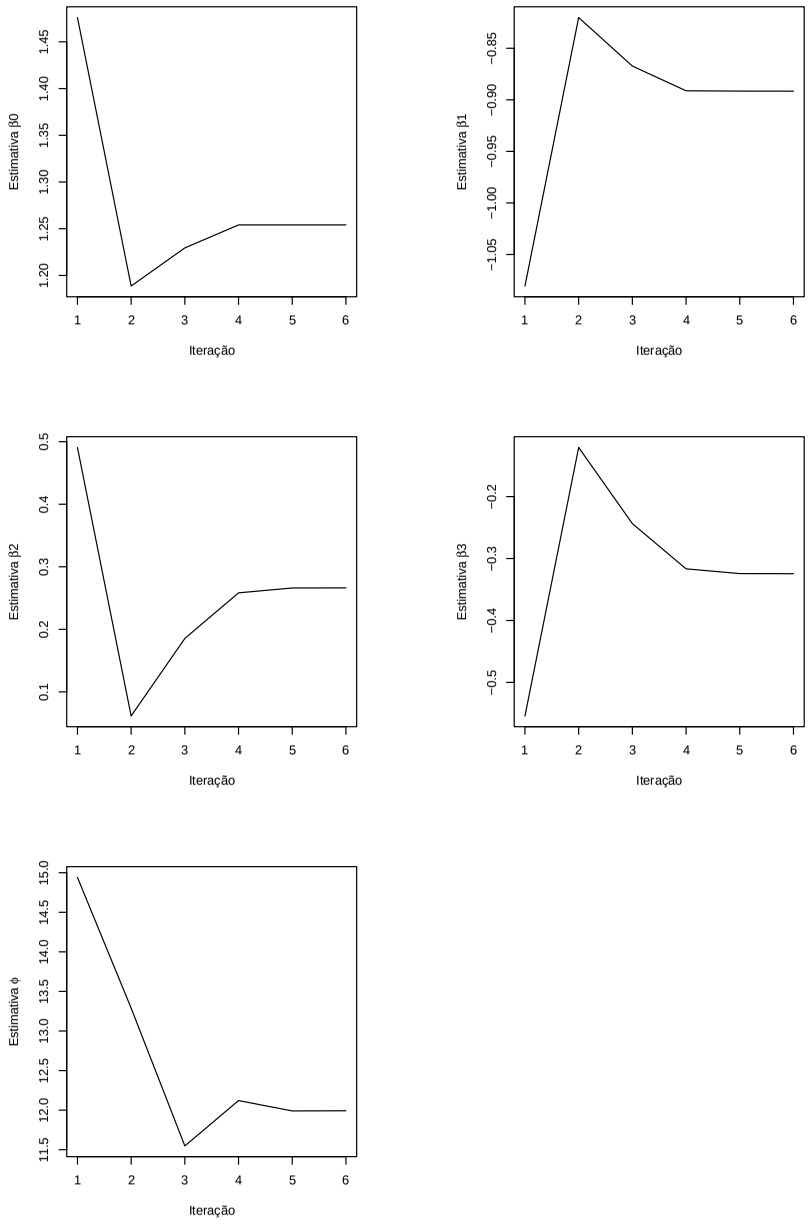

Figura 5.21: Comportamento das estimativas dos parâmetros em cada iteração do método de Newton para o conjunto de dados de QI sem pontos de alavanca.

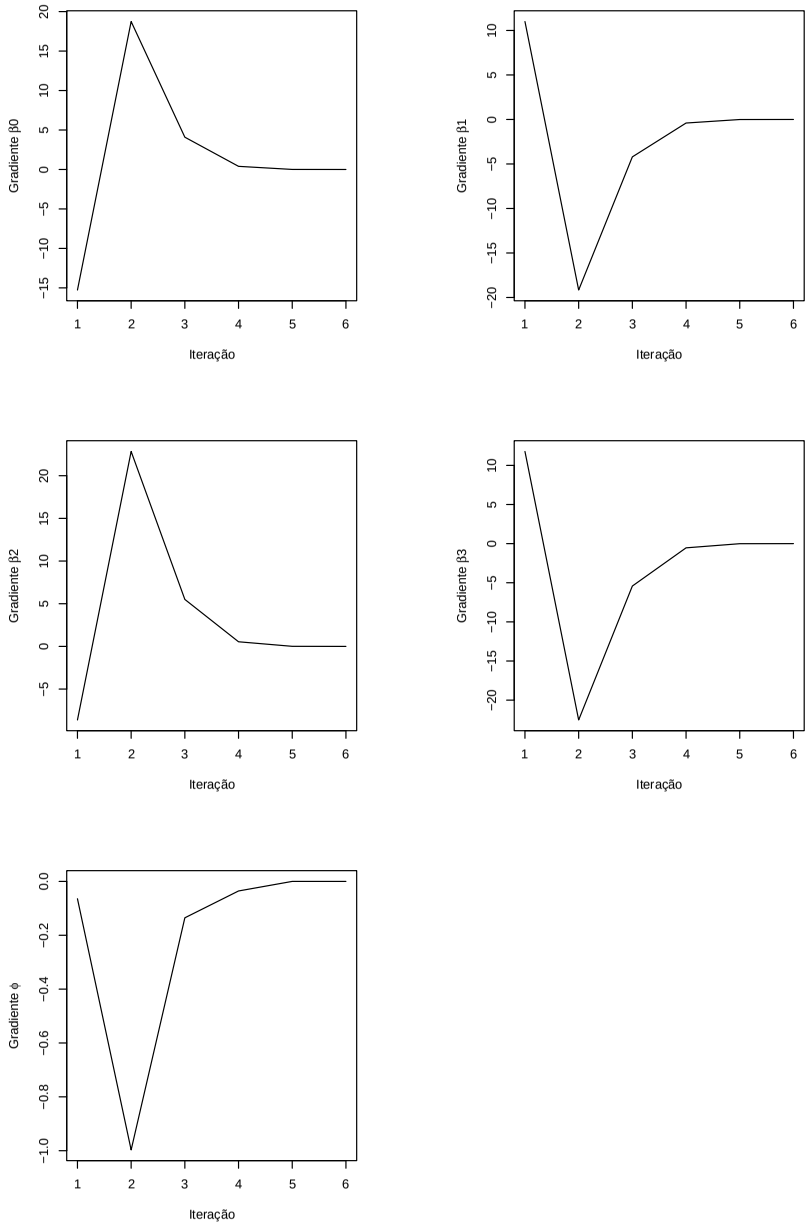

Figura 5.22: Comportamento dos gradientes em cada iteração do método de Newton para o conjunto de dados de QI.

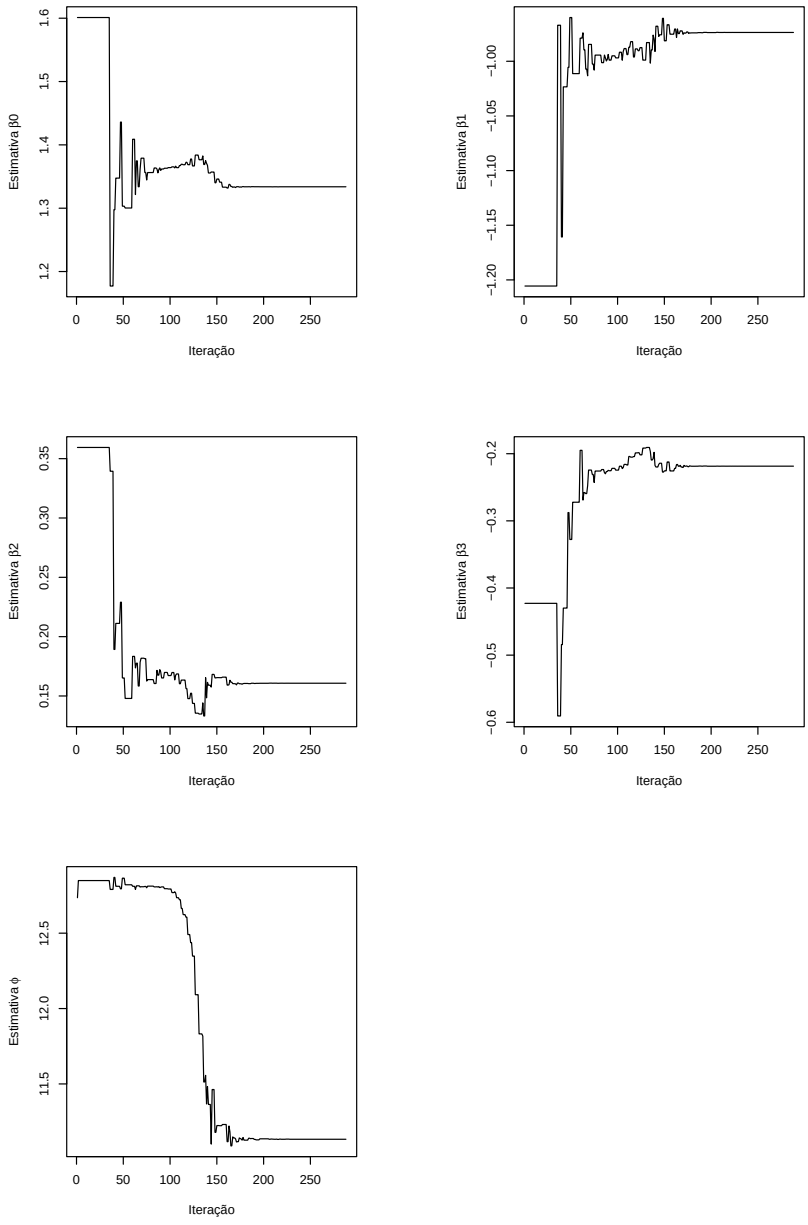

Figura 5.23: Comportamento das estimativas dos parâmetros em cada iteração do método simplex para o conjunto de dados de QI.

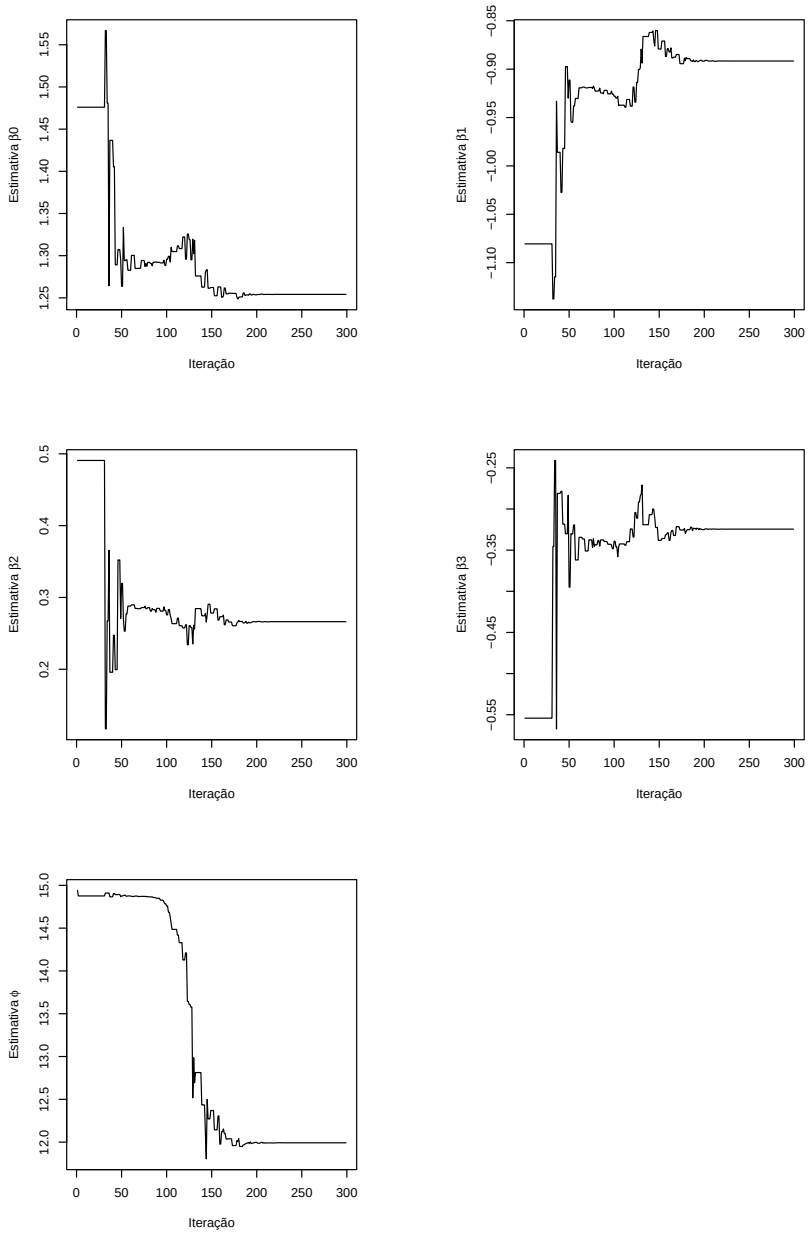

Figura 5.24: Comportamento das estimativas dos parâmetros em cada iteração do método simplex para o conjunto de dados de QI sem pontos de alavanca.

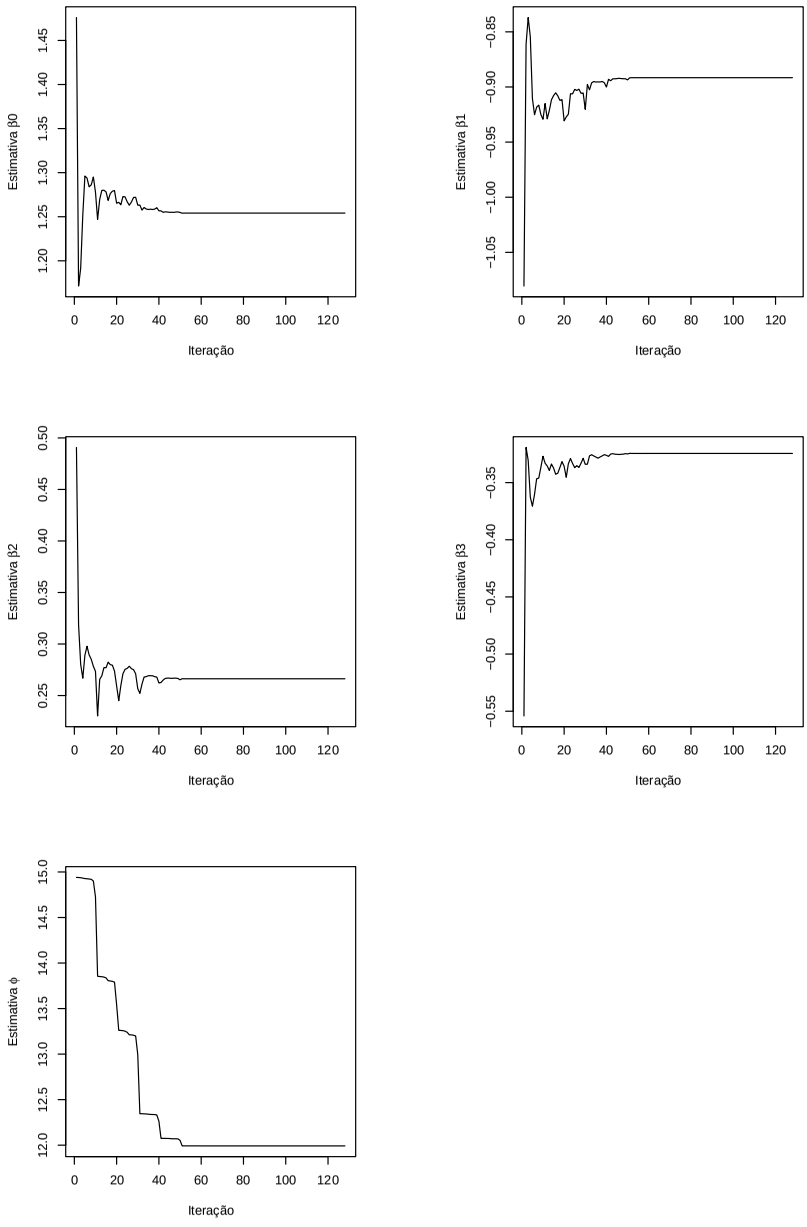

Figura 5.25: Comportamento das estimativas dos parâmetros em cada iteração do método gradiente conjugado para o conjunto de dados de QI.

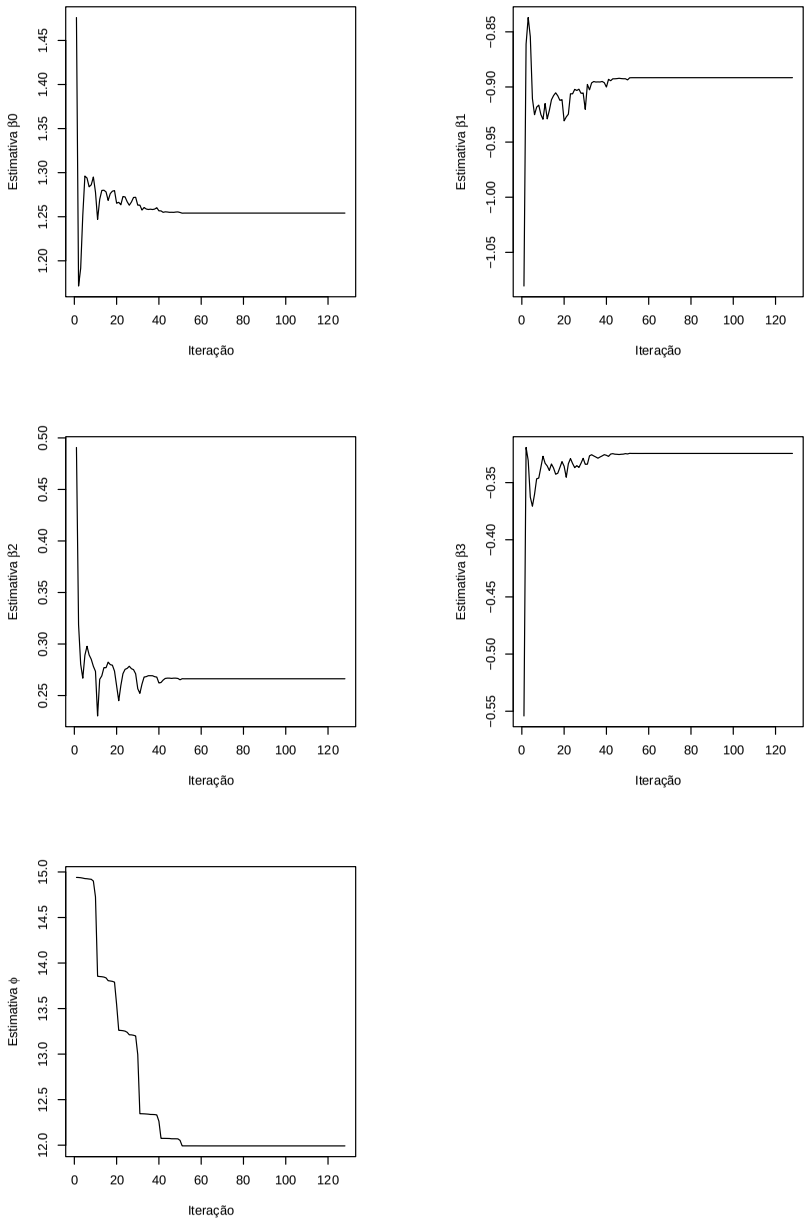

Figura 5.26: Comportamento das estimativas dos parâmetros em cada iteração do método gradiente conjugado para o conjunto de dados de QI sem pontos de alavanca.

CAPÍTULO 6

Conclusões

O presente trabalho avaliou a eficiência de alguns métodos de otimização não-linear como ferramentas a serem utilizadas na maximização numérica da função de log-verossimilhança do modelo de regressão beta. Considerando os resultados obtidos através das simulações de Monte Carlo realizadas e das aplicações apresentadas, pode-se concluir que os métodos de Newton e BFGS são os mais eficientes no que tange à estimação dos parâmetros do modelo.

Levando em conta critérios como número de iterações, taxa de convergência e taxa de erro, verificamos que os métodos que não usam informação referente à matriz hessiana da função de log-verossimilhança, como é o caso dos métodos simplex e gradiente conjugado, não são eficazes relativamente aos demais.

Todos os métodos forneceram estimativas semelhantes para os parâmetros do preditor linear. Os métodos BFGS e Newton sempre atingem o mesmo ponto, com a diferença de que o último necessita de um número menor de iterações. O método gradiente conjugado foi, em todos os casos considerados, o mais ineficiente tanto em termos de taxas de erro (sempre as maiores) quanto em termos de número de iterações e taxa de convergência.

Constatou-se que o maior obstáculo para todos os métodos estudados reside na estimação do parâmetro de precisão ϕ do modelo, pois, para alguns conjuntos de dados, a curvatura da função de log-verossimilhança tende a ser pouco acentuada na região próxima à do ponto de máximo, fazendo com que pequenas variações nesse parâmetro não resultem em variações significativas no valor da função a ser maximizada. Um outro aspecto que pode impactar a eficácia dos métodos é o ponto inicial usado no processo de otimização; os métodos BFGS e Newton não são afetados pela escolha desse ponto, ao passo que os métodos simplex e gradiente conjugado podem fornecer estimativas diferentes dependendo da distância do ponto inicial relativamente ao maximizador verdadeiro.

Na presença de pontos de alavanca, deve-se ter precaução em relação ao método de otimização empregado, pois alguns deles, como o simplex e o gradiente conjugado, podem apresentar desempenho falho em amostras de tamanho pequeno. Nesse sentido, o método de Newton foi eficiente, pois a magnitude dos pontos de alavanca não impactou significativamente nem o número de iterações executadas para atingir o ponto ótimo nem o comportamento das estimativas e gradientes da função na maximização da função de log-verossimilhança. Analogamente, o desempenho do método BFGS também não foi fortemente afetado por pontos atípicos.

Quando as covariáveis do modelo são substancialmente correlacionadas, o método de Newton continua sendo o mais eficiente, pois o tempo gasto no processo de otimização continua sendo o menor de todos para qualquer grau de correlação. Em grandes amostras, o método simplex apresentou leve tendência de aumento do tempo empregado quando a magnitude da correlação aumentou.

Considerando os resultados obtidos neste trabalho, de forma agregada, sugerimos que o método de Newton seja utilizado na estimação por máxima verossimilhança em modelos de regressão beta.

No que tange a trabalhos futuros, sugerimos a adoção de uma reparametrização do parâmetro de precisão ϕ que facilite sua estimação.

APÊNDICE

Programas de Simulação

Neste apêndice apresentamos os programas de simulação utilizados neste trabalho. Os programas foram desenvolvidos na linguagem de programação matricial `Ox` em sua versão `4.02` para a avaliação dos métodos de otimização BFGS, Newton e Simplex. O software `R` versão `2.1.1` foi empregado na avaliação do método de otimização gradiente conjugado. Estes programas fornecem as estimativas dos parâmetros do modelo de regressão beta proposto por Ferrari & Cribari-Neto (2004). Adicionalmente, eles calculam critérios de avaliação com base na precisão de estimativas, tais como erro quadrático médio (EQM), erro absoluto médio (MAE) e erro absoluto médio percentual (MAPE).

A.1 Programa de Simulação no Ox

```
/*************************************************************************
** PROGRAMA: simula.ox
**
** USO: Calcula estimativa e variancia dos parametros do
**      modelo de regressao beta alem do Erro Quadratico Medio (EQM),
**      Erro Medio Absoluto (MAE) e Erro Medio Percentual Absoluto
**      (MAPE) para os tamanhos amostrais n=20,40,60,80,100,200 e 500.
**
** AUTOR: Programa originalmente criado por Ferrari & Cribari-Neto (2004),
**          posteriormente modificado por Nataly Jimenez Monroy
**
** DATA INICIO: Maio 25/2006
**
** ULTIMA MODIFICACAO: Novembro 13/06
**
*************************************************************************/

/* Arquivos de cabecalho */
#include<oxstd.h>
#include<oxprob.h>
#import<maximize>
#include<oxdraw.h>

/* Variaveis globais */
static decl s_vy;
static decl s_mX;
decl yfunc;

/* Funcao de log-verossimilhanca */
 floglik(const vP, const adFunc, const avScore, const amHess)
 {
   decl k = rows(vP) - 1;
   decl eta = s_mX*vP[0:(k-1)];
   decl mu = exp(eta) ./ (1.0+exp(eta));
   decl phi = vP[k];
   decl ynew = log( yfunc ./ (1.0-yfunc) );
   decl munew = polygamma(mu*phi, 0) - polygamma((1.0-mu)*phi, 0);
   decl T = diag( exp(eta) ./ (1.0+exp(eta)) .^2 );

   adFunc[0] = double( sumc( loggamma(phi) - loggamma(mu*phi)
               - loggamma((1-mu)*phi) + (mu*phi-1) .* log(yfunc)
           + ( (1-mu)*phi-1 ) .* log(1-yfunc) ));
```

```
   if(avScore)
   {
      (avScore[0])[0:(k-1)] = phi*s_mX'*T*(ynew-munew);
      (avScore[0])[k] = double(sumc( polygamma(phi, 0) - mu .*
         polygamma(mu*phi, 0) - (1.0-mu) .* polygamma( (1.0-mu)*phi, 0) +
         mu .* log(yfunc) + (1.0-mu) .* log(1.0-yfunc) ));
   }

   if( isnan(adFunc[0]) || isdotinf(adFunc[0]) )
    return 0;

   else
        return 1;
 }

 /* Funcao que gera observacoes da normal multivariada */
 rmnorm(const n, const media, const sigma)
{
    decl y,lambda,d,w;
    eigensym(sigma,&lambda,&d);
    lambda=diag(lambda);
    y=rann(columns(sigma),n);
    w=media+d*lambda.^(1/2)*y;
    return(w');
}

main()
 {
  decl mu, phi, k, ir1,vp1, dfunc1, ybar, yvar, betaols, betaols1,
       ynew, dExecTime, nobs, olserrorvar, olsfitted, falha,
       olsfittednew, p, pr, qr, n, theta, beta0, beta1, beta2,
       etar, phir, mur, cs, REP, s_vX, estim, mae, mape, eqm, nmin,
       nmax, betaini, mmle, mvar, mmae, mmape, meqm;

/* Inicio da contagem do tempo */
  dExecTime = timer();
  ranseed("GM");

/* Tamanho minimo de amostra */
  nmin = 20;

/* Tamanho maximo de amostra */
  nmax = 100; //200,500
```

```
/* Numero de replicas da Simulacao */
  REP  = 1000;

/* Valores verdadeiros dos parametros */
  beta0 = 1.0;
  beta1 = 0.1;
  beta2 = 0.5;
  phir = 35;

/* Vetor de parametros */
  theta = beta0|beta1|phir;
//  theta = beta0|beta1|beta2|phir;

/* Numero de covariaveis */
  k = 1;

/* Inicializacao das matrices das replicas Monte Carlo */
  vp1 = zeros(rows(theta),REP);
  mmle = zeros(rows(theta),REP);
  mae = zeros(rows(theta),REP);
  mape = zeros(rows(theta),REP);
  eqm = zeros(rows(theta),REP);

/* Replicas de Monte Carlo */
  for(nobs = nmin; nobs <= nmax; nobs += 20)
    {
/* Geracao das covariaveis e dos parametros da funcao beta */
    decl media=(0|0);
    decl variancia=((1~0)|(0~1));
//  Amostra de observacoes da normal bivariada
//    s_vX=rmnorm(nobs,media,variancia) ;
//  Amostra de observacoes da normal univariada
     s_vX = rann(nobs, k);
//  Amostra de observacoes da normal univariada com pontos de alavanca
//   decl a, s_vX1;
//   s_vX1 = rann(nobs-1, k);
//   a = 2.5;//ponto de alavanca a=2.5,3.0,4.0,5.0,6.0 e 7.0
//   s_vX = s_vX1|a;
     s_mX = ones(nobs,1)~s_vX;
     p = columns(s_mX);
     etar =  theta[0:k][]' * s_mX';
     mur = exp(etar)./(1.0 + exp(etar));
/* Inicializacao da contagem de falhas*/
     falha = 0;
```

```
/* Geracao do vetor de respostas */
     for(cs = 0; cs < REP; ++cs)
      {
       decl cn;
       decl vY = zeros(nobs,1);
       for(cn = 0; cn < nobs; ++cn)
         {
          pr = (mur[cn] .* phir);
          qr = ((1 - mur[cn]) .* phir);
          vY[cn] = ranbeta(1,1,pr,qr);
         }
       if(vY[] >= 1.0 || vY[] <= 0.0)
        {
         println("\n\n ERRO: DADO FORA DO INTERVALO (0,1)!\n\n");
         exit(2);
        }

/* Vetor de respostas */
       s_vy = vY;

/* Vetor de respostas transformado */
       ynew = log(s_vy ./ (1.0 - s_vy));

/* Calculo do vetor inicial */
       if(p > 1)
        {
         ols2c(ynew, s_mX, &betaols1);
         betaols = betaols1;
        }

       else if(p == 1)
        {
         betaols = meanc(ynew);
        }

       olsfittednew = (s_mX)*betaols;
       olserrorvar = sumsqrc(ynew - olsfittednew)./(nobs-p);
       olsfitted = exp(olsfittednew) ./ (1.0 + exp(olsfittednew));
       ybar = meanc(s_vy);
       yvar = varc(s_vy);

/* Valores iniciais dos parametros */
//     vp1 = betaols|(meanc(1 ./ (olserrorvar'
```

```
//       *(olsfitted .* (1.0 - olsfitted)))) - 1.0);
//phi inicial sugerido por Ferrari & Cribari-Neto
     vp1 = betaols|15;  //phi inicial: 15,35,70
       betaini = vp1;
       yfunc = s_vy;

/* Controle de impressao de iteracoes */
// MaxControl(1000, 50);

/* Maximizacao da log-verossimilhanca */
//       ir1 = MaxBFGS(floglik, &vp1, &dfunc1, 0, FALSE);
       ir1 = MaxNewton(floglik, &vp1, &dfunc1, 0, TRUE);
//       ir1 = MaxSimplex(floglik, &vp1, &dfunc1, 0);

/* Calculo das medidas de precisao */
        if(ir1 == MAX_CONV || ir1 == MAX_WEAK_CONV)
         {
          mmle[][cs] = vp1;
          mae[][cs] = fabs(vp1 - theta);
          mape[][cs] = fabs((vp1 - theta) ./ theta)*100;
          eqm[][cs] = (vp1 - theta).^2;
         }
        else
          {
             ++falha;
             --cs;
             continue;
          }
   }

/* Resumo das medidas de precisao */
   estim = (meanr(mmle));
   mvar = (varr(mmle));
   mmae = (meanr(mae));
   mmape = (meanr(mape));
   meqm = (meanr(eqm));

/* Impressao dos resultados em arquivo */
    decl file =
    fopen("C:\\Users\\Alunos\\Nata\\Tese\\Resultados\\resultado.xls", "w");
    fprintln(file, "\n                RESUMO DAS MEDIDAS DE PRECISAO \n");
    fprintln(file, "\n Tamanho da amostra: ", nobs,
                   "\n Replicas com falha: ", falha,
                   "\n Replicas efectivas: ", (cs+falha),
```

```
                    "\n\n Beta real: ",theta,
                    "\n Beta inicial: ", betaini,
                    "\n Estimativas: ",estim,
                    "\n Variancia: ", mvar,
                    "\n EQM: ",meqm, "\n MAE: ",mmae,
                    "\n MAPE: ", mmape,
                    "\n Data: ", date(), "\n Hora: ", time(), "\n",
                    "\n Tempo total de execucao: \n", timespan(dExecTime));
    fclose(file);

/* Impressao dos resultados na tela */
//  println("\n ESTIMATIVA DOS PARAMETROS E RESUMO DAS MEDIDAS DE ERRO\n");
//  println("\n Tamanho da amostra: ", nobs,
//          "\n Replicas com falha: ", falha,
//          "\n Replicas efectivas: ", (cs+falha),
//          "\n\n Beta real ","Beta inicial ",theta~betaini);
//  println("%16.5f", "%c",
//            {"Estimativa","Variancia", "MAE", "MAPE","EQM"},
//          estim~mvar~mmae~mmape~meqm);
//  println("\n DATA: ", date(), "\n HORA: ", time(),
//          "\n", "\n TEMPO TOTAL DE EXECUCAO: \n", timespan(dExecTime));
//
/* Impressao do tempo */
//    print( "\n DATA: ", date() );
//    print( "\n HORA: ", time());
//    print( "\n TEMPO TOTAL DE EXECUCAO: ",timespan(dExecTime));
//    print( "\n" );
  }
 }
```

A.2 Programa de Simulação no R

```
#############################################################################
# PROGRAMA: sim_beta.txt
#
# USO: Calcula estimativa e variancia dos parametros do
#      modelo de regressao beta alem do Erro Quadratico Medio (EQM),
#      Erro Medio Absoluto (MAE) e Erro Medio Percentual Absoluto
#      (MAPE) para os tamanhos amostrais n=20,40,60,80,100,200 e 500.
#
# AUTOR: Nataly Jimenez Monroy
#
# DATA INICIO: Agosto 13/2006
#
# ULTIMA MODIFICACAO: Dezembro 01/06
#
#############################################################################

# Pacotes usados
library(betareg);
library(MASS);

rm(list=ls(all=TRUE))
n<-500  # Tamanho da amostra
rep<-1000 # Numero de replicas
# Valor verdadeiro dos parametros
b0<-1.0
b1<-0.1
#b2<-0.5
phir<-35

# Vetor de parametros
theta<-cbind(b0,b1,phir)
#theta<-cbind(b0,b1,b2,phir)

# Amostra da normal univariada
X1<-matrix(rnorm(n))

# Amostra da normal bivariada
Sigma <- matrix(c(1,0,0,1),2,2) # Matriz de correlacao
#X1<-mvrnorm(n, rep(0, 2), Sigma)

# Amostra com pontos de alavanca
#a<-7.0 # magnitude do ponto de alavanca a = 2.5,3.0,4.0,5.0,6.0,7.0
#X1<-c(Xa,a)
```

```
X<-cbind(1,X1)
p = ncol(X)
eta <- X%*%theta[1:p]
mu<-exp(eta)/(1+exp(eta))

# Inicializacao das matrizes de armazenamento
Y<-array(numeric(n*1),dim=c(n,1))
#coef<-array(numeric(rep*4),dim=c(rep,4))
coef<-array(numeric(rep*3),dim=c(rep,3))
#ape<-array(numeric(rep*4),dim=c(rep,4))
ape<-array(numeric(rep*3),dim=c(rep,3))
#ae<-array(numeric(rep*4),dim=c(rep,4))
ae<-array(numeric(rep*3),dim=c(rep,3))

# Funcao do pacote betareg para modificacao do algoritmo de otimizacao
# e do valor inicial do parametro phi
fix(br.fit)

# Contagem do tempo
tempo<-system.time(
for(i in 1:rep)
   {
   # Geracao do vetor de respostas
    for(j in 1:n)
       {
        pr<-mu[j]*phir
        qr<-(1-mu[j])*phir
        Y[j]<-rbeta(1,pr,qr)
       }

# Impresao da informacao do processo iterativo em arquivo
sink(file = "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\OutputR.txt",
#     append = TRUE, type = #c("output","message"),split = FALSE)

# Estimacao dos parametros
fitnorm<- betareg(Y~X-1)
sink()

# Armazenamento dos resultados em cada replica
coef[i,]<-betareg(Y~ X-1)$coef
ape[i,]<-abs(theta-coef[i,])/theta
ae[i,]<-abs(theta-coef[i,])
}
)
```

```
# Impressao dos resultados em arquivo
write.table(tempo, file =
            "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\simula.xls",
            sep = "    ", row.names = #FALSE, col.names = "TIEMPO",
            dec = ".", append = FALSE, qmethod = "double")

mediab0 <- mean(coef[,1])
mediab1 <- mean(coef[,2])
#mediab2 <- mean(coef[,3])
mediaphi <- mean(coef[,3])
#media<-t(cbind(mediab0,mediab1,mediab2,mediaphi))
media<-t(cbind(mediab0,mediab1,mediaphi))

write.table(media, file =
            "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\simula.xls",
            sep = "    ", row.names = #FALSE, col.names = "MEDIA",
            dec = ".", append = TRUE, qmethod = "double")

varb0<-var(coef[,1])
varb1<-var(coef[,2])
#varb2<-var(coef[,3])
varphi<-var(coef[,3])
#var<-t(cbind(varb0,varb1,varb2,varphi))
var<-t(cbind(varb0,varb1,varphi))

write.table(var, file =
            "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\simula.xls",
            sep = "    ", row.names = #FALSE, col.names = "VAR",
            dec = ".", append = TRUE, qmethod = "double")

eqmb0<-mean((ae[,1])^2)
eqmb1<-mean((ae[,2])^2)
#eqmb2<-mean((ae[,3])^2)
eqmphi<-mean((ae[,3])^2)
#eqm<-t(cbind(eqmb0, eqmb1, eqmb2, eqmphi))
eqm<-t(cbind(eqmb0, eqmb1, eqmphi))

write.table(eqm, file =
            "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\simula.xls",
            sep = "    ", row.names = #FALSE, col.names = "EQM",
            dec = ".", append = TRUE, qmethod = "double")

maeb0<-mean(ae[,1])
```

```
maeb1<-mean(ae[,2])
#maeb2<-mean(ae[,3])
maephi<-mean(ae[,3])
#mae<-t(cbind(maeb0, maeb1, maeb2, maephi))
mae<-t(cbind(maeb0, maeb1, maephi))

write.table(mae, file =
            "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\simula.xls",
            sep = "    ", row.names = #FALSE, col.names = "MAE",
            dec = ".", append = TRUE, qmethod = "double")

mapeb0<-mean(ape[,1])*100
mapeb1<-mean(ape[,2])*100
#mapeb2<-mean(ape[,3])*100
mapephi<-mean(ape[,3])*100
#mape<-t(cbind(mapeb0, mapeb1, mapeb2, mapephi))
mape<-t(cbind(mapeb0, mapeb1, mapephi))

write.table(mape, file =
            "c:\\Users\\Alunos\\Nata\\Tese\\Resultados\\simula.xls",
            sep = "    ", row.names = #FALSE, col.names = "MAPE",
            dec = ".", append = TRUE, qmethod = "double")

# Impressao de resultados na tela
tempo
media
var
eqm
mae
mape
```

Referências Bibliográficas

Berndt, E., Hall, B., Hall, R. & Hausman, J. (1974), 'Estimation and inference in nonlinear structural models', *Annals of Economic and Social Measurement* **3/4**, 653–665.

Buckley, J. (2003), 'Estimation of models with beta distributed dependent variables: A replication and extension of Paolino (2001)', *Political Analysis* **11**(1), 1–12.

Denbo, R., Eisenstat, S. & Steihaug, T. (1982), 'Inexact Newton methods', *SIAM Journal on Numerical Analysis* **19**, 400–408.

Doornik, J. & Ooms, M. (2005), An introduction to Ox, Technical report, `http://www.doornik.com/ox/OxIntro.pdf`.

Espinheira, P. L., Ferrari, S. L. P. & Cribari-Neto, F. (2006), 'On beta regression residuals', Technical report, Universidade de São Paulo.

Everitt, B. (1987), *Introduction to Optimization Methods and their Application in Statistics*, London: Chapman & Hall.

Ferrari, S. & Cribari-Neto, F. (2004), 'Beta regression for modelling rates and proportions', *Journal of Applied Statistics* **31**, 799–815.

Fletcher, R. & Reeves, C. (1964), 'Function minimization by conjugate gradients', *Computer Journal* **7**, 148–154.

Goldfarb, D. (1980), 'Curvilinear path steplength algorithms for minimization which use directions of negative curvature', *Mathematical Programming* **18**, 31–40.

Harville, D. A. (1997), *Matrix Algebra from a Statistician's Perspective*, New York: Springer.

Johnson, N., Kotz, S. & Balakrishnan, N. (1995), *Continuous Univariate Distributions*, 2nd edn, New York: John Wiley & Sons.

Khuri, A. (1993), *Advanced Calculus with Applications in Statistics*, New York: John Wiley & Sons.

Kieschnick, R. & McCullough, B. (2003), 'Regression analysis of variates observed in (0,1): Percentages, proportions and fractions', *Statistical Modelling* **3**, 193–213.

McCullagh, P. & Nelder, J. (1989), *Generalized Linear Models*, 2nd edn, New York: Chapman and Hall.

Montgomery, D., Peck, E. & Vining, G. (2001), *Introduction to Linear Regression Analysis*, New York: John Wiley & Sons.

Moré, J. & Sorensen, D. (1979), 'On the use of directions of negative curvature in a modified Newton method', *Mathematical Programming* **16**, 1–20.

Nelder, J. & Mead, R. (1965), 'A simplex algorithm for function minimization', *Computer Journal* **7**, 308–313.

Nocedal, J. & Wright, S. J. (1999), *Numerical Optimization*, New York: Springer-Verlag.

Ospina, R., Cribari-Neto, F. & Vasconcellos, K. (2006), 'Improved point and interval estimation for a beta regression model', *Computational Statistics and Data Analysis* **51**, 960–981.

Pammer, K. & Kevan, A. (2004), The contribution of visual sensitivity, phonological processing and non-verbal IQ to children's reading, Technical report, The Australian National University, Canberra.

Paolino, P. (2001), 'Maximum likelihood estimation of models with beta-distributed dependent variables', *Political Analysis* **9**, 325–346.

Prater, N. (1956), 'Estimate gasoline yields from crude', *Petroleum Refiner* **35**, 236–238.

Press, W., Teukolsky, S., Vetterling, W. & Flannery, B. (1992), *Numerical Recipes in C: The Art of Scientific Computing*, 2nd edn, New York: Cambridge University Press.

Rao, C. R. (1973), *Linear Statistical Inference and its Applications*, 2nd edn, New York: John Wiley & Sons.

Schnabel, R. & Eskow, E. (1991), 'A new modified Cholesky factorization', *SIAM Journal on Scientific Computing* **11**, 1136–1158.

Sherman, A. (1978), 'On Newton-iterative methods for the solution of systems of nonlinear equations', *SIAM Journal on Numerical Analysis* **15**, 755–771.

Smithson, M. & Verkuilen, J. (2006), 'A better lemon squeezer? Maximum-Likelihood regression with beta-distributed dependent variables', *Psychological Methods* **11**, 54–71.

Vasconcellos, K. & Cribari-Neto, F. (2005), 'Improved maximum likelihood estimation in a new class of beta regression models', *Brazilian Journal of Probability and Statistics* **19**, 13–31.

Venables, W. & Smith, D. (2004), *An Introduction to R*, R Development Core Team.