Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Estatística

Estudo e Diagnóstico de Transformações Box-Cox com Aplicações em Paralelização

Guilherme Alexandre Alvarez

Projeto apresentado para obtenção do título de Bacharel em Estatística.

Brasília
2015

GUILHERME ALEXANDRE ALVAREZ

Estudo e Diagnóstico de Transformações Box-Cox com Aplicações em Paralelização

Projeto apresentado para obtenção do título de Bacharel em Estatística.

Orientador: Prof. **Eduardo Monteiro de Castro Gomes**

Brasília
2015

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Estatística

Estudo e Diagnóstico de Transformações Box-Cox com Aplicações em Paralelização

por

Guilherme Alexandre Alvarez

Monografia apresentada ao Departamento de Estatística da Universidade de Brasília, como parte dos requisitos para obtenção do grau de

BACHAREL EM ESTATÍSTICA

Brasília, 15 de 06 de 2015.

Banca Examinadora:

Prof. Dr. Eduardo M. C. Gomes - Orientador (EST/UnB)

Prof. Dr. Donald Matthew Pianto - Membro (EST/UnB)

Prof. PhD Antônio Eduardo Gomes - Membro (EST/UnB)

DEDICATÓRIA

Dedico este trabalho à minha mãe,
pelo amor que sempre me proporcionou
e pelo apoio incondicional.
À você o meu amor e reconhecimento eterno...

Agradecimentos

À minha família por todo amor, consideração, incentivo e apoio incondicional.

Ao professor Eduardo Monteiro, pela orientação, conselhos e principalmente pela amizade.

À todos os professores do departamento de estatística.

Aos amigos e colegas do curso de estatística, em especial ao Felipe Quintino, Felipe Veloso, Alex Barros, Thiago Macedo, Agda Jéssica, Érica Rios, Geiziane Oliveira, Gustavo Martins, Nelson Oliveira e Juliano Santanna,

À todas as pessoas que contribuíram de alguma forma para a realização deste trabalho.

Resumo

Neste trabalho foi realizado um estudo sobre a família de transformações proposta por Box e Cox (1964). Por meio de estudos de simulação foram avaliados três intervalos de confiança e implementado técnicas de diagnóstico para valores influentes para o parâmetro de transformação. Devido à intensa computação necessária nos estudos de simulação, foi utilizada e avaliada computação em paralelo em R para diminuir o tempo total de computação.

Palavras-chave:Transformação Box-Cox, Diagnóstico, Simulação, Computação paralelo em R.

Abstract

In this work it is presented a study on the power transformation family proposed by Box and Cox (1964). By the use of simulations, the performance of three confidence intervals were evaluated and diagnostic techniques for influential values were implemented for the power transformation parameter. Because of the intensive computation required in the simulation studies, parallel computing in R was used to decrease the total amount of computing time.

Key-Words: Box-Cox transformation, Deletion diangnostics, Bootstrap, Parallel computation R.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

Análise de regressão é uma técnica estatística amplamente utilizada para descrever a relação entre uma variável resposta e uma ou mais variáveis explicativas. Atualmente o modelo de regressão de mínimos quadrados, que assume que os erros possuem distribuição normal com variância constante, é o mais difundido e utilizado devido a sua extensa aplicação e fácil implementação.

Quando um dos pressupostos do modelo de mínimos quadrados é violado, torna-se necessário a utilização de algum procedimento visando à correção desta falha. Por exemplo, quando a suposição de normalidade não é satisfeita, pode-se utilizar algum tipo de transformação para tentar normalizá-la ou ajustar outro modelo de regressão que assuma outra distribuição de probabilidade para os erros.

Apesar de existirem vários modelos que acomodam outras distribuições de probabilidade para os erros e com diferentes pressupostos, transformações não lineares ainda são artifícios comumente utilizados para melhorar a adequação do modelo de mínimos quadrados aos dados.

Este trabalho tem como objetivo realizar um estudo sobre transformações não lineares, especificamente a família de transformações propostas por Box e Cox (1964). Por meio de simulações são avaliados os desempenhos de três métodos de obtenção de intervalos de confiança para o parâmetro de transformação. Também é implementado o afastamento da verossimilhança descrita Cook e Weisberg (1982) como medida de influência e comparado com outras técnicas de diagnóstico tradicionalmente utilizados em análise de regressão.

Para a obtenção dos intervalos de confiança são considerados métodos de verossimilhança e de bootstrap e para o método de diagnóstico é considerada metodologia de deleção de casos. Como essas técnicas requerem computação intensiva torna-se interessante realizá-las em paralelo de modo a diminuir o seu tempo de execução.

Este trabalho está organizado de forma que na seção 2 é apresentada uma revisão bibliográfica dos principais conceitos abordados, na seção 3 é descrito os materiais e metodologia dos estudos de simulação, e os resultados são apresentados na seção 4.

2 Revisão Bibliográfica

2.1 Análise de Regressão

Em muitos aspectos análise de regressão está no centro da estatística. A sua extensa aplicação em áreas médicas, econômicas, psicológicas entre várias outras tornam análise de regressão talvez a técnica estatística mais utilizada. É um termo amplo para um conjunto de modelos usados para descrever a relação entre uma variável resposta (também chamada de dependente) e uma ou mais variáveis de explicativas (também chamadas independentes ou covariáveis).

2.1.1 Regressão de Mínimos Quadrados

Regressão de mínimos quadrados é um método de regressão em que os parâmetros do modelo são estimados minimizando a soma dos quadrados dos erros. Quando os pressupostos da regressão de mínimos quadrados são satisfeitos, os parâmetro estimados são não viesados e de variância mínima.

O modelo de regressão de mínimos quadrados com p variáveis independentes é dado por:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip} + \varepsilon_i, \quad i = 1, \ldots, n$$

Onde Y_i é a variável resposta, X_{ip} são as variáveis explicativas independentes assumidas fixas, β_p os parâmetros estimados e ε_i o erro.

Utilizando a notação matricial temos:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

onde

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \ \mathbf{X} = \begin{pmatrix} X_1^T \\ X_2^T \\ \vdots \\ X_n^T \end{pmatrix} = \begin{pmatrix} 1 & X_{11} & \cdots & X_{1p} \\ 1 & X_{21} & \cdots & X_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & \cdots & X_{np} \end{pmatrix}, \ \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \ \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

E os pressupostos do modelo são:

1. Os erros $\boldsymbol{\varepsilon}$ são não correlacionados.

2. Os erros $\boldsymbol{\varepsilon}$ são normalmente distribuídos com média 0.

3. Os erros $\boldsymbol{\varepsilon}$ possuem variância constante σ^2, ou seja, $\boldsymbol{\varepsilon} \sim N(0, \sigma^2 \mathbf{I}_n)$.

2.2 Transformações Não Lineares

Em muitos casos os pressupostos do modelo de mínimos quadrados, linearidade, normalidade e homocedasticidade, não são satisfeitos. Nessas situações, transformações não lineares da variável resposta frequentemente possibilitam a aplicação do modelo de mínimos quadrados.

2.2.1 Transformação Box e Cox

Em 1964 Box e Cox propuseram a seguinte família de transformações não-lineares:

$$y^{(\lambda)} = \begin{cases} \frac{y^{\lambda}-1}{\lambda}, & \text{se} \quad \lambda \neq 0, \ y > 0 \\ \log \ y, & \text{se} \quad \lambda = 0, \ y > 0 \end{cases}$$

O parâmetro λ pode ser estimado maximizando a função de log verossimilhança perfilada dada por (mais detalhes em como obtê-la vide apêndice):

$$\mathcal{L}_p(\lambda) = \mathcal{L}(\lambda/\mathbf{X}, \mathbf{Y}, \hat{\beta}, \hat{\sigma^2}) = -\frac{n}{2}\log(2\pi) - \frac{n}{2} - \frac{n}{2}\log(\hat{\sigma^2}) + (\lambda - 1)\sum_{i=1}^{n}\log(Y_i),$$

no qual

$$\hat{\sigma^2} = \frac{1}{n}(\mathbf{Y}^{\lambda} - \mathbf{X}\boldsymbol{\beta})^{\mathbf{T}}(\mathbf{Y}^{\lambda} - \mathbf{X}\boldsymbol{\beta})$$

A função de log verossimilhança perfilada estima $\hat{\lambda}$ que torna os resíduos o mais próximo da normalidade medido pela distância de Kullback-Leibler. Isto, no entanto, não garante que os resíduos são o suficientemente próximos da distribuição Normal para permitir o uso da regressão de mínimos quadrados, portanto é sempre necessário verificar a distribuição dos resíduos após a transformação da variável.

Uma vez estimado, $\hat{\lambda}$ é tido como conhecido e a análise de regressão é feita da maneira usual. Apesar de haver um custo em não saber o valor real de λ ele não é, em geral, grande o suficiente para descreditar o uso da técnica. É importante ressaltar que uma vez que a variável for transformada todas as interpretações sobre o modelo devem ser baseadas na variável transformada não na variável original.

Existem várias famílias de transformações não lineares e extensos estudos foram feitos sobre elas. Sakia (1992) e Li (2005) fornecem excelentes revisões sobre o assunto.

2.3 Intervalos de Confiança

Intervalos de confiança permitem descrever o grau de incerteza associado com a estimação de um parâmetro populacional a partir de uma amostra. Neste trabalho serão avaliados 3 tipos de intervalos para o parâmetro de transformação λ: Razão de Verossimilhanças, Wald e Bootstrap descritos abaixo.

2.3.1 Razão de Verossimilhanças

Considere um modelo estatístico com função de verossimilhança $\mathcal{L}(\theta|\mathbf{y})$ onde $\mathbf{y}$ é um vetor de uma amostra aleatória simples de uma distribuição $f(y|\theta)$. Considere as hipóteses nula H_0 e alternativa H_1. Seja $\mathcal{L}_0(\theta|\mathbf{y})$ a função de verossimilhança assumindo H_0 verdadeira e seja $\mathcal{L}_1(\theta|\mathbf{y})$ a função de verossimilhança assumindo H_1 verdadeira. A forma geral da razão de verossimilhanças é dado por:

$$\Lambda = \frac{\sup \mathcal{L}_0(\theta|\mathbf{y})}{\sup \mathcal{L}_1(\theta|\mathbf{y})}$$

Defina $D = -2\log(\Lambda)$. Então, sob certas condições de regularidade, D possui distribuição assintoticamente χ^2. Portanto um intervalo com $(1-\alpha)100\%$ de confiança é dado por: $c = \boldsymbol{\theta} = (\theta_1, \dots) : D < \chi^2(1-\alpha)$

2.3.2 Wald

O intervalo de Wald é um intervalo baseado na distribuição assintótica dos estimadores de máxima verossimilhança $\hat{\boldsymbol{\theta}}$. Sob certa condições de regularidade, assintoticamente $\hat{\boldsymbol{\theta}}$ tem distribuição Normal multivariada com vetor de médias $\boldsymbol{\theta}$ e matriz de variâncias e covariâncias dada pelo inverso da informação de Fisher, ou seja:

$$\hat{\boldsymbol{\theta}} \xrightarrow{d} N_k(\boldsymbol{\theta}, I^{-1}),$$

no qual, I^{-1} é o inverso da informação de Fisher e k é a dimensão de $\hat{\boldsymbol{\theta}}$. Sob um ponto de vista computacional I^{-1} corresponde ao inverso da matriz Hessiana. Assim um intervalo com $(1-\alpha)100\%$ de confiança é dado por: $\hat{\theta} \pm z_{\alpha/2}\sqrt{I^{-1}}$

O intervalo de Wald é mais popular do que o de razão de verossimilhanças devido à sua simplicidade e fácil obtenção. Apesar de eles serem assintoticamente equivalentes, em pequenas amostras eles podem diferir o suficiente para levar a conclusões diferentes.

O intervalo de razão de verossimilhanças é preferível ao de Wald porque ele utiliza apenas uma aproximação (de que a estatística do teste é chi quadrado) enquanto o que o de Wald utiliza duas (de que a estatística do teste é normal e da variância estimada).

2.3.3 Bootstrap

O método de Bootstrap consiste em obter uma distribuição empírica de uma estatística através de repetidas amostras aleatórias com reposição obtidas da amostra original. Bootstrap permite gerar intervalos de confiança e testar hipóteses sem ter de assumir uma distribuição teórica subjacente específica para a estatística de interesse.

Considere uma amostra de tamanho n retirada de uma população e seja $\hat{\theta}$ um estimador para o parâmetro θ de interesse. Retirando-se R amostras com reposição da amostral original, calcula-se o estimador $\hat{\theta}$ para cada amostra bootstrap. Os estimadores de cada amostra boostrap serão denominadas $\hat{\theta}^*_r$. Sejam $\hat{\theta}^*_{(1)}, \dots, \hat{\theta}^*_{(R)}$ os valores

ordenados dos estimadores $\hat{\theta}_r^*$. Um intervalo de confiança percentílico para θ é dado por:

$$\hat{\theta}^*_{((R+1)\alpha/2)} < \theta < \hat{\theta}^*_{((R+1)(1-\alpha/2))}$$

Apesar de ser muito intuitivo e simples de calcular, o método percentílico não considera o viés e assimetria na distribuição empírica do bootstrap. Para corrigir este problema, Efron (1987) propôs os intervalos percentílicos acelerados ajustado de viés e assimetria (BC_a em inglês). Os seguintes passos são necessários para obter um intervalo BC_a de θ:

- Calcule:

$$\hat{z} = \Phi^{-1}\left[\frac{\underset{r=1}{\overset{R}{\#}}(\hat{\theta}_r^* \leq \hat{\theta})}{R+1}\right]$$

  onde Φ^{-1} é o quantil da Normal padrão e $\#(\hat{\theta}_r^* \leq \hat{\theta})/R+1$ é a proporção de $\hat{\theta}_r^*$ menores ou iguais à estatística $\hat{\theta}$ calculada da amostra original. Se a distribuição do bootstrap for simétrica e se $\hat{\theta}$ for não viesado, então a proporção será próxima de 0.5 e o fator de correção $\hat{z}$ será próximo de zero.

- Seja $\hat{\theta}_{(-i)}$ o valor da estatística $\hat{\theta}$ quando a i-ésima observação é deletada da amostra. Seja $\overline{\hat{\theta}}$ a média de $\hat{\theta}_{(-i)}$, i.e. $\sum_{i=1}^n \hat{\theta}_{(-i)}/n$, então calcule:

$$a = \frac{\sum_{i=1}^n (\hat{\theta}_{(-i)} - \overline{\hat{\theta}})^3}{6\left[\sum_{i=1}^n (\hat{\theta}_{(-i)} - \overline{\hat{\theta}}^2)\right]^{\frac{3}{2}}}$$

- Com os fatores de correção a e $\hat{z}$ calcule:

$$\begin{aligned} z_L &= \Phi\left[\hat{z} + \frac{\hat{z} - z_{1-\alpha/2}}{1 - a(\hat{z} - z_{1-\alpha/2})}\right] \\ z_U &= \Phi\left[\hat{z} + \frac{\hat{z} + z_{1-\alpha/2}}{1 - a(\hat{z} + z_{1-\alpha/2})}\right] \end{aligned}$$

onde Φ é a distribuição acumulada da Normal padrão. Assim os valores z_L e z_U são utilizados para calcular o intervalo percentílico BC_a dado por:

$$\hat{\theta}^*_{Rz_L} < \theta < \hat{\theta}^*_{Rz_U}$$

2.4 Técnicas de Diagnóstico

O parâmetro λ estimado pela função de verossimilhança no entanto, é sensível à valores influentes. Uma transformação pode ser sugerida a um modelo cujos resíduos possuem distribuição aproximadamente Normal devido à alguns valores discrepantes por exemplo. Da mesma forma, uma transformação pode não ser sugerida a um modelo cujos resíduos não possuem distribuição Normal. Em qualquer caso, o modelo estaria comprometido e conclusões errôneas poderiam ser obtidas. Desta maneira faz-se importante identificar quais valores influenciam a estimação de λ.

Neste trabalho será considerado o afastamento da verossimilhança como medida de influência para λ bem como outras técnicas de diagnóstico tradicionalmente utilizadas em análise de regressão.

2.4.1 Afastamento da Verossimilhança

Utilizando deleção de casos, Cook e Weisberg (1982) desenvolveram um método de análise de influência denominado afastamento da verossimilhança. Seja $\mathcal{L}_p(\hat{\theta})$ o valor da máxima log verossimilhança e seja $\mathcal{L}_p(\hat{\theta}_{(i)})$ o valor da log verossimilhança no ponto $\hat{\theta}_{(i)}$, ou seja, o valor da log verossimilhança no ponto de estimação após a deleção da i-ésima observação. O afastamento da verossimilhança é dado por:

$$LD_i = 2[\mathcal{L}_p(\hat{\theta}) - \mathcal{L}_p(\hat{\theta}_{(i)})]$$

O afastamento da verossimilhança pode ser visto como um teste de razão de verossimilhanças (TRV), onde $H_0 : \theta = \hat{\theta}_{(i)}$. Ou seja, é verificado se ao deletar uma observação, o novo valor $\hat{\theta}_{(i)}$ é estatisticamente igual ao $\hat{\theta}$ estimado com os dados completos. Assintoticamente, o TRV possui distribuição χ^2_q, onde q é a dimensão de θ. Desta maneira, se o p-valor calculado for menor do que o nível de confiança α, então a i-ésima observação é considerada influente.

É pertinente ressaltar que o p-valor gerado pelos TRVs são aproximações devido ao fato de que o TRV possui distribuição assintótica χ^2_q e não distribuição exata χ^2_q, consequentemente esses valores são apenas aproximações.

Uma das dificuldades na implementação do método é a necessidade de reestimação dos parâmetros após a deleção de cada caso o que pode gerar um custo computacional alto, principalmente se for considerada a deleção de mais de uma observação. Devido à esta dificuldade e à pequena capacidade computacional disponível à época, Cook e Wang (1983) e posteriormente Tsai e Wu (1992), propuseram aproximações de um passo do afastamento da verossimilhança para λ.

2.4.2 Alavancagem

Considere a matriz $\mathbf{H}$ definida como: $\mathbf{H_{nxn}} = \mathbf{X}(\mathbf{X^T X})^{-1}\mathbf{X^T}$. Os elementos da diagonal principal desta matriz, denotados por h_{ii} são chamados de alavancagem da i-ésima observação. Eles são utilizados para verificar se os valores de X da i-ésima observação são discrepantes porque pode ser demonstrado que h_{ii} é uma medida da distância entre os valores de X da i-ésima observação e a média de X para todas as observações. Portanto, um alto valor de h_{ii} indica que a i-ésima observação está distante do centro das observações X.

Um valor de alavancagem h_{ii} geralmente é considerado alto se for duas vezes maior do que a média das alavancagens, denotado por $\overline{h}$ e calculado como:

$$\overline{h} = \frac{\sum_{i=1}^{n} h_{ii}}{n}$$

Para um conjunto de dados razoavelmente grande, com relação ao número de parâmetros do modelo, h_{ii} é considerado alto a partir de 0.5

2.4.3 Resíduos

Uma das técnicas de diagnóstico muito utilizada para detecção de valores discrepantes para Y é a análise de resíduos. O resíduo para a i-ésima observação é obtido através de $e_i = y_i - \hat{y}_i$, que calcula a diferença entre o valor observado e o valor ajustado. Estes resíduos, no entanto, não são muito informativos por não apresentar variância constante, $Var(e_i) = \sigma^2(1 - h_{ii})$, que depende de h_{ii}. Desta forma, uma maneira mais apropriada de comparar os resíduos é padronizá-los, então obtém-se o resíduo padronizado dado por:

$$e_i^* = \frac{e_i}{\sqrt{\sigma^2(1 - h_{ii})}}$$

Como o resíduo de cada observação não é independente da variância estimada, não se obtém um distribuição t-Student como seria esperado. Para contornar este problema substitui-se σ^2 por $\sigma^2_{(i)}$, o erro quadrático médio do modelo sem a i-ésima observação. O índice (i) indica que a i-ésima observação foi excluída. Assim o resíduo studentizado é dado por:

$$t_i = \frac{e_i}{\sqrt{\sigma^2_{(i)}(1 - h_{ii})}} \sim t_{n-p-1}$$

Assim, uma observação é considerada um outlier se o seu resíduo studentizado for maior do que o quantil da distribuição $t_{n-p-1}1 - \alpha$. No entanto, calcular o resíduo studentizado para todas as observações aumenta a probabilidade de erro tipo I. Assim é necessário utilizar a correção de Bonferroni - α/n em vez de α - quando o teste for realizado.

2.4.4 Distância de Cook

A distância de Cook é uma medida de diagnóstico para avaliar o impacto de uma observação na estimação dos parâmetros de regressão do modelo. A distância de Cook, que depende de h_{ii} e de e_i, é dada por:

$$D_i = \frac{e_i^2}{p\sigma}\left(\frac{h_{ii}}{(1-h_{ii})^2}\right)$$

Assim como para h_{ii}, quando o número de observações é grande em relação ao número de parâmetros, D_i é considerado alto se for maior do que 0.5.

3 Metodologia

3.1 Estudo de Simulação I

O primeiro estudo de simulação foi realizado para comparar os desempenhos dos três métodos de obtenção de intervalos de confiança para o parâmetro λ. Os intervalos foram avaliados de acordo com a sua cobertura nominal (proporção de intervalos que contém o verdadeiro valor do parâmetro) e a sua amplitude média. Um coeficiente de 95% foi fixado para a obtenção dos intervalos.

Foram considerados dois valores comumente utilizados para λ: 0.5 (raíz quadrada) e 0 (logaritmo). Foram ajustados modelos de regressão com uma covariável com distribuição Poisson (100) e o fator erro com distribuição Normal (0, σ). Para verificar se os intervalos de confiança são afetados pela variância do modelo, foram considerados dois valores para σ: 5 e 10. Em cada caso foram geradas 1000 amostras distintas com tamanhos 30 e 100. Para cada modelo gerado foram aplicadas as transformações exponencial e quadrática e em seguida estimado o parâmetro λ e os intervalos de confiança. Para o método bootstrap foram consideradas 1000 reamostras com reposição para cada amostra gerada.

O pacote *boot* do ambiente R de programação foi utilizado para realizar o estudo bootstrap e o pacote *car* que possui as funções *powerTransform* e *testTransform* foi utilizado para os outros intervalos.

3.2 Estudo de Simulação II

O segundo estudo de simulação foi realizado para implementar o afastamento da verossimilhança com mais de um valor influente como medida de diagnóstico e verificar como as observações influentes se comportam sob as outras técnicas de diagnóstico.

Para a realização do estudo, foi construído um modelo com três variáveis explicativas geradas a partir de amostras provenientes de distribuições Uniforme (0, 100), Exponencial (0.5) e Poisson (500), além do componente erro com distribuição Normal (0, 20). Para cada variável foi fixado os parâmetros $\beta_1 = 1, \beta_2 = 2$ e $\beta_3 = 0.1$ respectivamente. O tamanho amostral foi de 400 observações.

A escolha das distribuições, os parâmetros e o tamanho amostral foram escolhidos de modo que o modelo possa representar um conjunto de dados real e que o custo computacional seja alto para evidenciar as vantagens da paralelização.

Assim o modelo construído foi:

$$Y = X_1 + 2X_2 + 0.1X_3 + \varepsilon,$$

onde

- $X_1 \sim$ Unif(0, 100)

- $X_2 \sim$ Exp(0.5)
- $X_3 \sim$ Pois(500)
- $\varepsilon \sim$ N(0, 20)

O modelo estimado a partir das variáveis explicativas foi o seguinte:

$$Y = -12.9806 + 1.003X_1 + 2.224X_2 + 0.127X_3,$$

Que possui R^2 ajustado: 0.669, erro padrão estimado: 20.34 e valor do teste F: 270.140 com 396 graus de liberdade.

Em seguida foram introduzidos duas observações influentes que foram obtidas fixando valores para as variáveis explicativas, e em seguida ajustando a variável resposta perfeitamente ao modelo estimado e somando cinco erros padrões.

Assim a primeira observação influente foi obtida fixando $X_1 = 105, X_2 = 13, X_3 = 569$ e o valor para Y foi obtido: -12.9806 + 1.003*105 + 2.224*13 + 0.127*569 + 5*20.34 = 295.343. O mesmo foi feito para a segunda observação influente $X_1 = 110, X_2 = 16, X_3 = 574$, a resposta Y = -12.9806 + 1.003*110 + 2.224*16 + 0.127*574 + 5*20.34 = 307.667

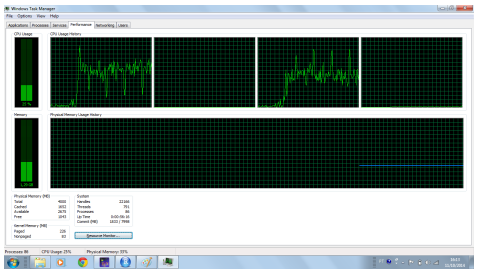
O ambiente R de programação possui as funções *powerTransform* e *testTransform* no pacote *car* que geram o estimador de máxima verossimilhança de λ e calcula testes de razão de verossimilhança para valores específicos de λ, respectivamente. Essas duas funções foram as principais utilizadas para implementar o método.

3.3 Paralelização

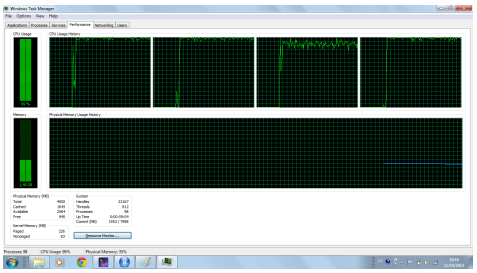
Computação em paralelo é uma forma de computação que consiste em dividir problemas grandes em tarefas menores, que então são resolvidas concorrentemente (em paralelo), com o objetivo de reduzir o tempo total de processamento.

Como a taxa de melhoria da frequência dos processadores se tornou mais lenta e custosa em termos de consumo de energia, o uso de computação paralela na forma de processadores multinúcleo está presente em praticamente todos os computadores atualmente.

R é uma linguagem e ambiente de desenvolvimento integrado gratuito, criado em 1993, para cálculos estatísticos e gráficos. Apesar de ser um software largamente utilizado pela comunidade estatística ele utiliza por default apenas um núcleo do computador, o que impossibilita a utilização da capacidade total do computador como pode ser observado pela figura abaixo.



(a) R utilizando apenas um núcleo por default



(b) R utilizando os quatro núcleos através de paralelização

Figura 1: Comparação entre processamentos no R, sem e com paralelização

As técnicas de bootstrap e deleção de casos utilizados na metodologia recaem na categoria de problemas embaraçosamente paralelizáveis (embarassingly parallel problems em inglês) que são problemas em que as tarefas são independentes e portanto não há necessidade de comunicação entre os núcleos o que as torna fáceis de paralelizar.

Para a realização do primeiro estudo de simulação em paralelo, foi utilizado as opções *parallel="snow"*, *ncpus*=4 da função *boot*. Para o segundo estudo de simulação foram utilizados os pacotes *doSNOW* e *foreach* que permitem utilizar paralelização explicitamente. Ambos os estudos foram feito no R versão 3.1.1. Para mais detalhes da programação utilizada, vide o apêndice.

Como no primeiro estudo de simulação foram considerados 8 cenários distintos, para medir o efeito da paralelização foi considerado apenas um cenário do estudo com $n = 100, \sigma = 10$ e $\lambda = 0$ e medidos os seus tempos de computação com e sem paralelização.

Todas as computações foram feitas em um computador com as seguintes especificações:

```
Sistema
Manufacturer: ASUSTeK Computer Inc.  Product Name: K84C

Placa-mãe

Processador: Intel(R) Core(TM) i3-2330M CPU @ 2.2 GHz, 2200 MHz, 2 Core(s),
4 logical Processor(s)
Manufacturer: Intel

OS: WIN7 Basic

CPU op-mode(s): 64-bit

Uso Atual da Memoria Fisica
     total    used     free
Mem: 3.91G    2.6G 1.  23GB
```

4 Resultados

Nesta seção são apresentados os resultados dos estudos de simulação descritos nas seções 3.1 e 3.2.

4.1 Resultados do Estudos de Simulação I

A tabela abaixo apresenta os coeficientes de coberturas nominais e amplitudes médias dos diferentes intervalos de confiança obtidos para cada caso simulado.

Tabela 1: Coberturas Nominais e (médias de amplitudes) de intervalos de confiança obtidos com os métodos: Wald, Razão de Verossimilhanças (RV) e Bootstrap (BC_a).

n	λ	σ	Wald	RV	BC_a
30	0.5	10	0.9600	0.9600	0.9610
			(2.0705)	(2.0585)	(2.2010)
30	0.5	5	0.9490	0.9530	0.9730
			(1.9984)	(1.9915)	(2.2342)
30	0	10	0.9540	0.9520	0.9600
			(0.0420)	(0.0421)	(0.04196)
30	0	5	0.949	0.9500	0.9677
			(0.0402)	(0.0408)	(0.0468)
100	0.5	10	0.9540	0.9500	0.9300
			(1.0178)	(1.0196)	(0.9978)
100	0.5	5	0.9490	0.9480	0.9430
			(0.9956)	(0.9999)	(1.0050)
100	0	10	0.965	0.9600	0.9389
			(0.0207)	(0.0207)	(0.0201)
100	0	5	0.9540	0.9550	0.9265
			(0.01975)	(0.01988)	(0.01989)

Através da tabela 1 verifica-se que, de maneira geral, os intervalos de Wald e RV tiveram desempenhos muito parecidos. O intervalo BC_a teve a performance mais fraca, tanto em amplitude quanto em cobertura. Desta forma, o intervalo de Wald se torna o mais vantajoso devido à sua facilidade de obtenção, dependendo apenas da estimação de λ e o seu erro padrão obtido através da matriz hessiana.

Ao contrário do que seria esperado, a variância do modelo parece ter pouca influência na obtenção dos intervalos, reduzindo apenas um pouco as amplitudes e aproximando as coberturas nominais ao nível de confiança fixado. O tamanho amostral no entanto, tem uma grande impacto nas amplitudes reduzindo-os substancialmente quando o tamanho amostral aumenta.

Outro fator que possui forte influência sobre a amplitude dos intervalos é a linearidade do modelo. Para o modelo logarítmico ($\lambda = 0$), a média das amplitudes dos intervalos foram cerca de 50 vezes menores do que para o modelo com raiz quadrada (λ=0.5).

4.2 Resultados do Estudo de Simulação II

Para o modelo antes da introdução de valores influentes, a transformação sugerida foi de 0.995 com intervalo de RV de 95% de confiança: $[0.822 \leq \lambda \leq 1.173]$, como era de se esperar, já que o modelo segue todos os pressupostos da regressão linear.

Já para o modelo com valores influentes, a transformação sugerida foi de 0.781 com intervalo de RV de 95% de confiança: $[0.642 \leq \lambda \leq 0.920]$. Neste cenário, o 1 está fora do intervalo de confiança e uma transformação teria que ser aplicada.

Abaixo estão os gráficos descritos na revisão bibliográfica para o modelo com valores influentes:

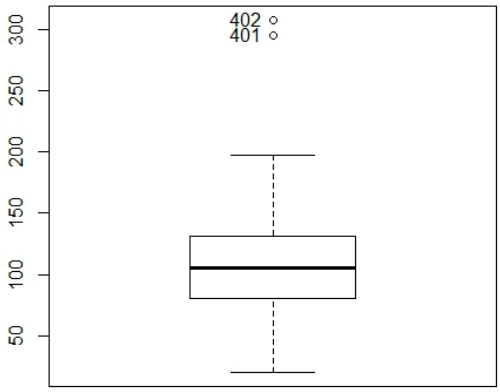


Figura 2: Boxplot da variável resposta Y

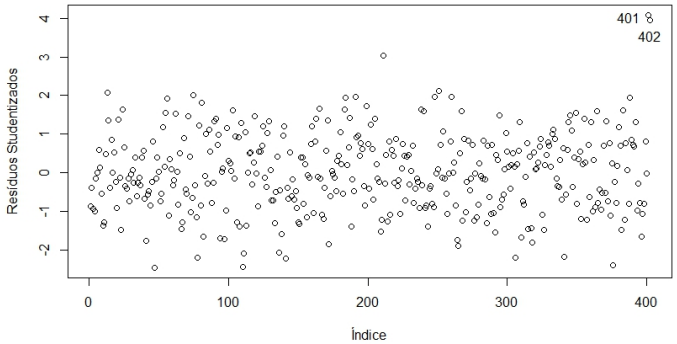


Figura 3: Gráfico de dispersão dos resíduos studentizados

Através do boxplot (Figura 2) da variável reposta, é possível perceber que as observações introduzidas são fortes candidatas a outliers. Analisando o gráfico de resíduos studentizados do modelo (Figura 3), é possível perceber que os resíduos das observações 401 e 402 são as que mais se distanciam de 0 apesar não ficar claro pelo gráfico se elas são de fato outliers. Ao realizar o test t de para detecção de outliers com correção de Bonferroni as observações 401 e 402 possuem p-valores de 0.022 e 0.037 respectivamente e podem ser consideradas outliers.

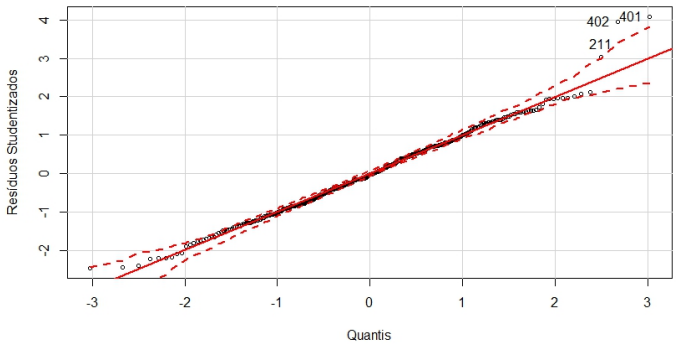


Figura 4: Gráfico de probabilidade normal com envelope de confiança de 95%

Através da Figura 4 é possível notar que os valores introduzidos estão distantes da normalidade e fora do envelope de confiança.

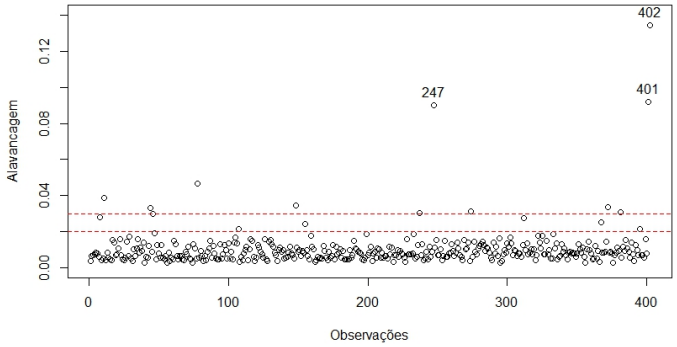


Figura 5: Gráfico de dispersão dos valores h_{ii} com linhas horizontais de 2 e 3 vezes $\overline{h}$

Ao examinar a Figura 5 verifica-se que as observações introduzidas possuem as maiores alavancagens, apesar de não serem muito distantes do resto das observações e consideradas baixas se 0.5 for adotado como valor de referência.

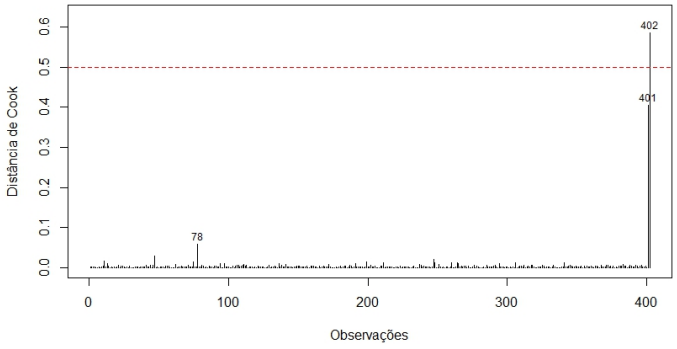


Figura 6: Gráfico da medida da distância de Cook com linha de referência de 0.5

De todos os gráficos, a distância de Cook é a que mais destaca os valores influentes, apesar de que apenas a observação 402 possa ser considerada influente se 0.5 for tomado como valor de referência.

Os métodos de diagnóstico tradicionais fornecem bons indícios de quais valores podem influenciar na estimação de λ, mas eles não são definitivos e a aplicação da análise de influência torna-se necessária.

Ao aplicar o afastamento da verossimilhança deletando os casos um a um, nenhum valor foi considerado influente. O menor p-valor encontrado foi de 0.324, obtido ao deletar a observação 402, bem distante do nível crítico 0.05. Isto se deve ao fato de que duas observações influentes foram incluídas no modelo, assim uma observação "mascara"o efeito da outra sob a estimação de λ. Além disso o esforço computacional

foi baixo, necessitando de apenas 5.98 segundos para realizar o método.

Desta maneira, uma alternativa para detectar as observações influentes é deletar os casos dois a dois, o que aumenta consideravelmente o custo computacional. Assim foram realizados $\binom{402}{2} = 80601$ testes de razão de verossimilhanças e o menor p-valor encontrado foi 0.0026 obtido justamente ao deletar as observações 401 e 402. Deste modo, conclui-se que as observações 401 e 402 são conjuntamente influentes na estimação de λ.

Apesar de o método não ter detectado as observações influentes deletando os casos um a um, é possível plotar o gráfico de afastamento de verossimilhança LD_i, proposto por Cook e Wang (1983):

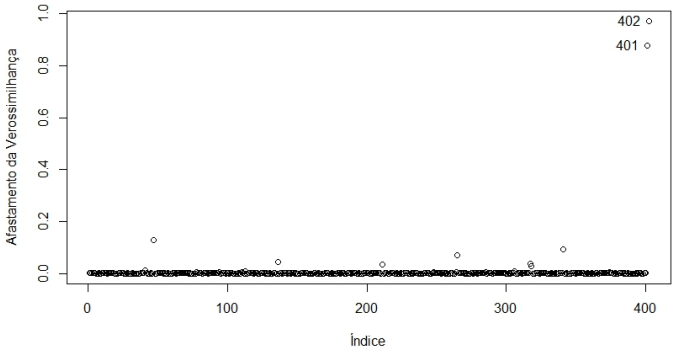
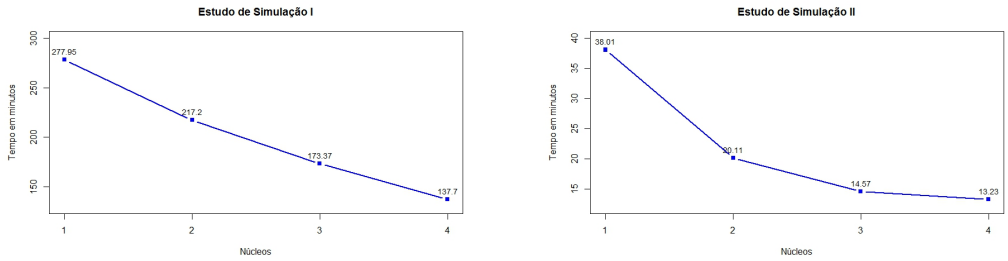


Figura 7: Gráfico de afastamento da verossimilhança LD_i

Analisando a Figura 7, percebe-se que, assim como o gráfico da distância de Cook, as observações 401 e 402 claramente se distanciam das demais.

4.3 Paralelização

A Figura abaixo descreve o tempo computacional gasto para a realização dos estudos de simulação variando a quantidade de núcleos utilizados.



(a) Tempo gasto para um cenário do primeiro estudo de simulação

(b) Tempo total gasto no segundo estudo de simulação

Figura 8: Gráfico do tempo total de computação para a realização dos estudos de simulação

Para um cenário do primeiro estudo de simulação, o tempo de execução diminuiu de 277.95 para 137.7 minutos ou um pouco menos da metade. Se essa redução for extrapolada para os outros 7 cenários, houve uma redução aproximada de 1122 minutos ou 18.7 horas do tempo total necessário para concluir o primeiro estudo de simulação.

Já para o segundo estudo, a redução foi de 38.01 para 13.23 minutos, um pouco mais de dois terços. Em ambos os casos a redução não foi de um quarto devido ao fato de que o R tem que alocar as tarefas para cada núcleo e reciprocamente juntar os resultados obtidos e deste modo gera um custo computacional extra.

5 Conclusão

Neste trabalho foi realizado um estudo sobre a família de transformações Box-Cox. O estudo de simulação para comparação de diferentes intervalos de confiança indicou que o intervalo de Wald e o de RV são muito parecidos, enquanto que o intervalo BC_a, que demanda computação intensiva, obteve o desempenho mais fraco. Assim, o intervalo de Wald é o mais vantajoso devido à sua simplicidade de obtenção.

O estudo de simulação para a análise de influência demonstrou a eficácia da técnica de diagnóstico desenvolvida por Cook e Weisberg (1983) para dados com mais de uma observação influente e mostrou como valores influentes para λ se comportam sob outras ferramentas de diagnóstico.

No primeiro estudo de simulação, a computação em paralelo reduziu em um pouco mais da metade o tempo total de execução. No segundo estudo de simulação a redução foi cerca de um terço do tempo total para a deleção dos casos dois a dois. Ambos os estudos evidenciam as vantagens da paralelização para problemas computacionalmente intensivos.

Pode-se considerar como possíveis trabalhos futuros os seguintes temas: aplicação da metodologia de diagnóstico para casos multivariados; utilização de paralelização em outros problemas embaraçosamente paralelizáveis como MCMC, análise de agrupamentos, floresta aleatória, entre outros.

Referências

[1] Box, G.E.P.; Cox, D.R. (1964). **An Analysis of Transformations**. Journal of the Royal Statistical Society. Series B (Methodological), Vol. 26, No. 2. (1964), pp. 211-252.

[2] Cook, R.D.; Weisberg, S. (1982). **Residuals and Influence in Regression**. School of statistics, university of Minnesota

[3] Atkinson, A. C. (1985). **Plots, Transformations, and Regression** Oxford.

[4] Cook, R.D.; Wang, P.C. (1983). **Transformation and Influential Cases in Regression**. Tecnometrics, 1983, 25337-343

[5] Tsai, C.L.; WU, X. (1990). **Diagnostics in Transformation and Weighted Regression**. Tecnometrics, 1990, VOL 32, NO.3

[6] Neter, J.; Waserman, W.; Kutner, M.H. (1983) **Applied Linear Regression Models** Richard D. Irwin Inc, Homewood Illinois 6043,1983

[7] Li, P. (2005). **Box-Cox Transformations: An Overview** Department of Statistics, University of Connecticut

[8] Sakia, R.M. (1992). **The Box-Cox transformation technique: a review** Sokoine University of Agriculture, Department of Crop Science and Production, Box 3005, Morogoro, Tanzania

[9] Gomes, E. M. C. (2013). **Modelos Rathie-Swamee: Aplicações e extensão para modelos de regressão** Universidade de São Paulo, Escola Superior de Agricultura Luiz de Queiroz, Piracicaba, 2013.

[10] Rosario, R. (2010). **Taking R to the Limit, Part I: Parallelization**. Los Angeles R User Group, 2010.

[11] John Fox and Sandford Weisberg **An R Companion to Applied Regression**. Sage, Thousand Oaks CA. URL http://socserv.socsi.mcmaster.ca/jfox/Books/Companion

[12] Bradley Efron. **Better bootstrap confidence intervals**. Journal of the American Statistical Association, 82(397):171?185, March 1987

[13] John Fox (2002) **Bootstrapping Regression Models Appendix to An R and S-PLUS Companion to Applied Regression**. Sage, Thousand Oaks CA. URL http://socserv.socsi.mcmaster.ca/jfox/Books/Companion

[14] Revolution Analytics and Steve Weston (2014). **doSNOW: Foreach parallel**. adaptor for the snow package. R package version 1.0.12. http://CRAN.R-project.org/package=doSNOW

[15] R Core Team (2014). **R a language and enviroment for statistical computing**. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

6 Apêndices

6.1 Obtenção da log verossimilhança perfilada de λ

Assumindo que as variáveis Y^λ possuem distribuição normal e que elas são independentes, então a densidade conjunta de $\mathbf{Y}^\lambda$ pode ser escrita como o produto das densidades de Y^λ:

$$\begin{aligned} f(\mathbf{Y}^\lambda) &= \prod_{i=1}^{n} f(Y_i^\lambda) \\ &= \prod_{i=1}^{n} (2\pi\sigma^2)^{(-\frac{1}{2})} \exp\left[-\frac{1}{2\sigma^2}(Y_i^\lambda - X_i^T\boldsymbol{\beta})^2\right] \\ &= (2\pi\sigma^2)^{(-\frac{n}{2})} \exp\left[-\frac{1}{2\sigma^2}(\mathbf{Y}^\lambda - \mathbf{X}\boldsymbol{\beta})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\boldsymbol{\beta})\right] \end{aligned}$$

O jacobiano da transformação de $\mathbf{Y}$ para $\mathbf{Y}^\lambda$ é dado por $J(\lambda, \mathbf{Y}) = \frac{d\mathbf{Y}^\lambda}{d\mathbf{Y}} = \prod_{i=1}^{n} \mathbf{Y}^{\lambda-1}$, assim a densidade para $\mathbf{Y}$, que também é a função de verossimilhança para o modelo completo, é dado por:

$$\mathcal{L}(\lambda, \beta, \sigma/\mathbf{X}, \mathbf{Y}) = f(\mathbf{Y}) = (2\pi\sigma^2)^{(-\frac{n}{2})} \exp\left[-\frac{1}{2\sigma^2}(\mathbf{Y}^\lambda - \mathbf{X}\boldsymbol{\beta})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\boldsymbol{\beta})\right] \prod_{i=1}^{n} Y_i^{\lambda-1}$$

E a sua log verossimilhança:

$$\begin{aligned} \log(\mathcal{L}(\lambda, \beta, \sigma/\mathbf{X}, \mathbf{Y})) &= -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}(\mathbf{Y}^\lambda - \mathbf{X}\boldsymbol{\beta})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\boldsymbol{\beta}) + \\ &\quad (\lambda - 1)\sum_{i=1}^{n} \log(Y_i) \end{aligned}$$

Tomando $\boldsymbol{\beta}$ e σ como os EMV's

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= (\mathbf{X}\mathbf{X}')^{-\mathbf{1}}\mathbf{X}'\mathbf{Y}^\lambda \\ \hat{\sigma}^2 &= \frac{(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})}{n} \end{aligned}$$

Podemos substituí-los na equação de verossimilhança, assim é possível obter a função de log verossimilhança perfilada para λ:

$$\begin{aligned}
\mathcal{L}(\lambda/\mathbf{X}, \mathbf{Y}, \hat{\beta}, \hat{\sigma}^{\mathbf{2}}) &= -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\hat{\sigma}^2) - \frac{1}{2\hat{\sigma}^2}(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}}) + \\
&\quad (\lambda - 1)\sum_{i=1}^{n}\log(Y_i) \\
&= -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\hat{\sigma}^2) - \frac{1}{2/n}\left(\frac{(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})}{(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})^{\mathbf{T}}(\mathbf{Y}^\lambda - \mathbf{X}\hat{\boldsymbol{\beta}})}\right) + \\
&\quad (\lambda - 1)\sum_{i=1}^{n}\log(Y_i) \\
&= -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\hat{\sigma}^2) - \frac{n}{2} + (\lambda - 1)\sum_{i=1}^{n}\log(Y_i)
\end{aligned}$$

Seja g a média geométrica do vetor da variável resposta $\mathbf{Y}$ (i.é., $g = (\prod_{i=1}^{n} Y_i)^{\frac{1}{n}}$) e seja $\mathbf{Y}(\lambda, g) = \frac{\mathbf{Y}^\lambda}{g^{\lambda-1}}$ então fica fácil de ver que:

$$\mathcal{L}(\lambda/\mathbf{X}, \mathbf{Y}(\lambda, g), \hat{\beta}, \hat{\sigma}^2) = C - \frac{n}{2}\log(s_\lambda^2),$$

em que s_λ^2 é a soma de quadrados do resíduo divido por n do ajuste do modelo linear $\mathbf{Y}(\lambda, g) \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I}_n)$. Então para maximizar a função de log verossimilhança perfilada, basta encontrar λ que minimize:

$$s_\lambda^2 = \frac{(\mathbf{Y}(\lambda, g) - \mathbf{X}\hat{\boldsymbol{\beta}})^{\mathbf{T}}(\mathbf{Y}(\lambda, g) - \mathbf{X}\hat{\boldsymbol{\beta}})}{n}$$

6.2 Programação utilizada no R:

```
###ESTUDO DE SIMULAÇÃO I
library(car)
library(boot)

set.seed(10)

###Função adaptada da função 'estimateTransform' do pacote car
VER <- function (object, lambda = rep(1, dim(object$y)[2])){
  fam <- match.fun(object$family)
  Y <- cbind(object$y)
  nc <- dim(Y)[2]
  nr <- nrow(Y)
  lam <- if (length(lambda) == 1)
    rep(lambda, nc)
  else lambda
  xqr <- object$xqr
  w <- if (is.null(object$weights))
    1
  else sqrt(object$weights)
  llik <- function(lambda) {
    (nr/2) * log(((nr - 1)/nr) * det(var(qr.resid(xqr, w *
                                 fam(Y, lam, j = TRUE)))))
  }
  LR <- llik(lambda)
  LR
}

f <- function(lambda, ralt, e1, e2, e3){
  maxVer <- powerTransform(ralt ~ e1, method="BFGS")$value
  potencia <- powerTransform(ralt ~ e1, method="BFGS")
  VER(potencia, lambda) - (maxVer + qchisq(.95, 1)/2)
}

###Função adaptada da função 'estimateTransform' do pacote car
estimate <- function (X, Y, weights = NULL, family = "bcPower", start = NULL,
                      method = "BFGS", ...) {
  fam <- match.fun(family)
  Y <- as.matrix(Y)
  X <- as.matrix(X)
  w <- if (is.null(weights))
    1
  else sqrt(weights)
  nc <- dim(Y)[2]
  nr <- nrow(Y)
  xqr <- qr(w * X)
```

```
  llik <- function(lambda) {
    (nr/2) * log(((nr - 1)/nr) * det(var(qr.resid(xqr, w *
                                     fam(Y, lambda, j = TRUE)))))
  }
  llik1d <- function(lambda, Y) {
    (nr/2) * log(((nr - 1)/nr) * var(qr.resid(xqr, w * fam(Y,
                                                 lambda, j = TRUE))))
  }
  if (is.null(start)) {
    start <- rep(1, nc)
    for (j in 1:nc) {
      res <- suppressWarnings(optimize(f = function(lambda) llik1d(lambda,
      Y[, j, drop = FALSE]), lower = -3, upper = +3))
      start[j] <- res$minimum
    }
  }
  res <- optim(start, llik, hessian = TRUE, method = method, ...)
  if (res$convergence != 0)
    warning(paste("Convergence failure: return code =", res$convergence))
  res$start <- start
  res$lambda <- res$par
  names(res$lambda) <- if (is.null(colnames(Y)))
    paste("Y", 1:dim(Y)[2], sep = "")
  else colnames(Y)
  roundlam <- res$lambda
  stderr <- sqrt(diag(solve(res$hessian)))
  lamL <- roundlam - 1.96 * stderr
  lamU <- roundlam + 1.96 * stderr
  structure(list(wald.inf = lamL, wald.sup = lamU))
}

###Função adaptada da função 'powerTransform' do pacote car
power <- function (object, data, subset, weights, na.action, ...) {
  mf <- match.call(expand.dots = FALSE)
  m <- match(c("object", "data", "subset", "weights", "na.action"),
             names(mf), 0L)
  mf <- mf[c(1L, m)]
  mf$drop.unused.levels <- TRUE
  mf[[1L]] <- as.name("model.frame")
  names(mf)[which(names(mf) == "object")] <- "formula"
  mf <- eval(mf, parent.frame())
  mt <- attr(mf, "terms")
  y <- model.response(mf, "numeric")
  w <- as.vector(model.weights(mf))
  if (is.null(w))
    w <- rep(1, dim(mf)[1])
```

```
  if (is.empty.model(mt)) {
    x <- matrix(rep(1, dim(mf)[1]), ncol = 1)
  }
  else {
    x <- model.matrix(mt, mf)
  }
  estimate(x, y, w, ...)
}

lambda <- function(formula, data, indices) {
  library(car)
  d <- data[indices, ]
  fit <- powerTransform(formula, data=d, method="BFGS")
  return(coef(fit))
}


tamanhoVer <- numeric()
tamanhoBoot <- numeric()
tamanhoWald <- numeric()
ICinfVer <- numeric()
ICsupVer <- numeric()
ICinfBoot <- numeric()
ICsupBoot <- numeric()
ICinfWald <- numeric()
ICsupWald <- numeric()
maxVer <- vector()
W <- vector()

n <- 100
sigma <- 10

set.seed(10)
tempoTotal <- system.time(
  for(i in 1:1000){
    e1 <- rpois(n, 100)
    error <- rnorm(n, 0, sigma)
    r <- e1 + error
    ralt <- exp(r)

    ###TRV
    ICinfVer[i] <- tryCatch(uniroot(f, c(-.9, 0), ralt=ralt, e1=e1)$root,
                  error=function(e){warning(conditionMessage(e)); NA})
    ICsupVer[i] <- tryCatch(uniroot(f, c(0, .9), ralt=ralt, e1=e1)$root,
                  error=function(e){warning(conditionMessage(e)); NA})
    tamanhoVer[i] <- as.numeric(ICsupVer[i])  - as.numeric(ICinfVer[i])
```

```
    ###WALD
    results <- power(ralt ~ e1)
    ICinfWald[i] <- results$wald.inf
    ICsupWald[i] <- results$wald.sup
    tamanhoWald[i] <- ICsupWald[i] - ICinfWald[i]


    ###BOOTSTRAP
    dados <- as.data.frame(cbind(ralt, e1))
    resultados <- boot(data=dados, statistic=lambda,
                              R=1000, formula=ralt ~ e1,
                              parallel="snow", ncpus=4)
    ICinfBoot[i] <- boot.ci(resultados, type="bca")$bca[4]
    ICsupBoot[i] <- boot.ci(resultados, type="bca")$bca[5]
    tamanhoBoot[i] <- ICsupBoot[i] - ICinfBoot[i]

    cat("\r", (i/1000)*100, "%")
  }
)

mean(tamanhoBoot, na.rm=T)
mean(tamanhoWald)
mean(tamanhoVer, na.rm=T)


coberNomBoot <- which(0>ICinfBoot & 0<ICsupBoot)
length(coberNomBoot)/1000

coberNomWald <- which(0>ICinfWald & 0<ICsupWald)

length(coberNomWald)/1000


coberNomVer <- which(is.na(tamanhoVer))
1 - length(coberNomVer)/1000

tempoTotal


tempos <- c(16677.24, 13032.02, 10401.92, 8262.08)
temposmin <- tempos/60
temposmin <- round(temposmin, digits=2)
indice <- 1:4
opar <- par(no.readonly=TRUE)
```

```
par(opar)
par(mar=c(5, 5, 5, 3))
par(mai=c(1.2, 1.2, 1, .2))
plot(temposmin ~ indice, type="b", main="Estudo de Simulação I",
     ylab="Tempo em minutos", xlab="Núcleos", pch=15, lty=1, lwd=2,
     col="blue", ylim=c(130,300), xaxt="n")
text(indice, temposmin, labels=temposmin, pos=3, cex=0.9)
#text(locator(4), c("38.01", "20.11", "14.57", "13.23"))
axis(side=1, at=c(1,2,3,4), col.axis="black")



###ESTUDO DE SIMULAÇÃO II
library(car)
library(doSNOW)
set.seed(10)
e1 <- runif(400,0,100)
e2 <- rexp(400,0.5)
e3 <- rpois(400,500)
error <- rnorm(400,0,20)
r <- e1+ 2*e2 + 0.1*e3 +error
ajuste1 <- lm(r~e1+e2+e3)
summary(ajuste1)
potencia1 <- powerTransform(r~e1+e2+e3)
summary(potencia1)
confint(ajuste1)


e1alt <- c(e1, 105, 110)
e2alt <- c(e2, 13, 16)
e3alt <- c(e3, 569, 574)
ralt <- c(r, (193.6433+5*20.34), (205.9673+5*20.34))
potencia2 <- powerTransform(ralt~e1alt+e2alt+e3alt)
summary(potencia2)
ajuste2 <- lm(ralt~e1alt+e2alt+e3alt)


###COM OUTLIER
boxplot(ralt)
names(ralt) <- seq(1:402)
identify(rep(1,402), ralt)



###COM
qqPlot(ajuste2,id.method="identify", xlab="Quantis",
```

```
       ylab="Resíduos Studentizados", cex=0.7, distribution="norm")


getAnywhere('plot.lm')
###COM
cutoff <- 4/(length(e1alt)-length(ajuste2$coefficients)-2)
plot(ajuste2, which=4, cook.levels=cutoff, caption="",sub.caption="")
abline(h=c(0.5,1), lty=2, col="red")
getAnywhere('plot.lm')
abline(h=cutoff, lty=2, col="red")



hat.plot <- function(fit) {
  p <- length(coefficients(fit))
  n <- length(fitted(fit))
  plot(hatvalues(fit),ylab="Alavancagem", xlab="Observações",
       ylim=c(0,0.14))
  abline(h=c(2,3)*p/n, col="red", lty=2)
  identify(1:n, hatvalues(fit), names(hatvalues(fit)))
}
###COM
hat.plot(ajuste2)



###COM
indice <- c(1:402)
plot(rstudent(ajuste2)~indice, xlab="Índice", ylab="Resíduos Studentizados")
identify(indice,rstudent(ajuste2))
outlierTest(ajuste2)



###Função adaptada da função 'testTransform' do pacote car
TRV <- function (object, lambda = rep(1, dim(object$y)[2])){
  fam <- match.fun(object$family)
  Y <- cbind(object$y)
  nc <- dim(Y)[2]
  nr <- nrow(Y)
  lam <- if (length(lambda) == 1)
    rep(lambda, nc)
  else lambda
  xqr <- object$xqr
  w <- if (is.null(object$weights))
```

```
    1
  else sqrt(object$weights)
  llik <- function(lambda) {
    (nr/2) * log(((nr - 1)/nr) * det(var(qr.resid(xqr, w *
                                                 fam(Y, lam, j = TRUE)))))
  }
  LR <- 2 * (llik(lambda) - object$value)
  LR
}
afastamento <- matrix()
for(i in 1:length(ralt)){
  afastamento[[i]] <- TRV(potencia2,
  coef(powerTransform(ralt[-i]~e1alt[-i]+ e2alt[-i] + e3alt[-i])))
}
afastamento
indice <- c(1:402)
plot(indice, afastamento, ylab = "Afastamento da Verossimilhança",
     xlab="Índice")
identify(indice, afastamento)

system.time(
pds <- foreach(i=1:length(ralt), .combine=c)%do%(
  (testTransform(potencia2, coef(powerTransform(ralt[-i]~e1alt[-i]+
  e2alt[-i]+e3alt[-i]))))$pval
)
)
pds
min(pds)


combs <- combn(402, 2)
cl <- makeCluster(4)
registerDoSNOW(cl)
system.time(
pv1 <- foreach(i=1:(length(combs)/2),.combine=c,.packages="car")%dopar%(
    (testTransform(powerTransform(ralt~e1alt+e2alt+e3alt),
              coef(powerTransform(ralt[-combs[,i]]~e1alt[-combs[,i]]+
                        e2alt[-combs[,i]]+e3alt[-combs[,i]]))))$pval
  )
)
stopCluster(cl)

which(pv1 < 0.05)
combs[, 80601]
pv1[80601]
```

```
tempos <- c(2280.72, 1206.80, 874.40, 793.61)
temposmin <- tempos/60
temposmin <- round(temposmin,digits=2)
indice <- 1:4
plot(temposmin~indice, type="b", ylab="Tempo em minutos", xlab="Núcleos",
pch=15, lty=1, lwd=2, col="blue", ylim=c(12,40), xaxt="n")
text(indice, temposmin, labels=temposmin, pos=3, cex=0.9)
axis(side=1, at=c(1,2,3,4), col.axis="black")
```