ESTATÍSTICA AVANÇADA: MODELOS NÃO LINEARES

Modulo 2: Transformações de Variáveis

Gilvan Guedes [Cedeplar - UFMG]
Melissa Pinho [Estatística - UFMG]

Escola do Legislativo - ALMG
Belo Horizonte, Minas Gerais

10 de setembro de 2015

Sumário

1 Breve Revisão de Regressão Linear

Neste módulo, estamos assumindo que você já tenha familiaridade com os princípios algébricos de regressão linear múltipla. Portanto, vamos apenas revisar conceitos fundamentais, pressupostos, e passar diretamente para a parte prática.

O modelo de regressão linear múltipla tenta modelar o relacionamento entre duas ou mais variáveis explicativas e a variável resposta. Sejam $X_1, X_2, \ldots, X_p$ as variáveis explicativas e Y a variável resposta. A equação do modelo é dada por:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \ldots + \beta_p x_{pi} + \epsilon_i \tag{1}$$

Os pressupostos para o modelo de regressão linear são:

1. $\epsilon_i \sim N(\mu = 0, \sigma^2)$
2. ϵ_i é i.i.d.
3. $E[X, \epsilon] = 0$
4. β é o vetor que projeta $\mathbf{X}$ linearmente em $\mathbf{Y}$
5. A matriz $\mathbf{X}$ tem posto completo

Em notação matricial, a Equação (1) pode ser escrita como:

$$\mathbf{Y} = \beta\mathbf{X} + \epsilon \tag{2}$$

onde $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_n)^T$, $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^T$, $\epsilon = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T$ e

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1p} \\ 1 & x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \ldots & x_{np} \end{bmatrix}$$

Para calcular os coeficientes estimados, $\hat{\beta}$, geralmente usa-se o Método de Mínimos Quadrados Ordinários (MQO), que consiste em minimizar a soma dos quadrados dos erros:

$$\begin{aligned} \min Q &= \textstyle\sum_{i=1}^{n} (\epsilon_i)^2 \\ \min Q &= \textstyle\sum_{i=1}^{n} [y_i - (\beta_0 \beta_1 y_{1i} + \beta_2 x_{2i} + \ldots + \beta_p x_{pi})]^2 \end{aligned} \tag{3}$$

No modelo acima, tal estimador é dado por:

$$\hat{\beta} = (\mathbf{X^T X})^{-1} (\mathbf{X^T Y}) \tag{4}$$

O estimador na Equação (4) é não viciado, ou seja, $E[\hat{\beta}] = \beta$, e possui variância dada por:

$$\mathbf{Var}(\hat{\beta}) = (\mathbf{X X^T})^{-1} \sigma^{\mathbf{2}} \tag{5}$$

O valor de $\sigma^{\mathbf{2}}$ pode ser estimado através do quadrado médio do resíduo, que é um estimador não viciado para $\sigma^{\mathbf{2}}$, ou seja, $E[\hat{\sigma^{\mathbf{2}}}] = \sigma^{\mathbf{2}}$. Sua fórmula é dada por:

$$\hat{\sigma}^{\mathbf{2}} = \frac{\hat{\epsilon}^{\mathbf{T}} \hat{\epsilon}}{n - p} \tag{6}$$

O coeficiente de determinação, R^2, mede a proporção da variabilidade da variável resposta $\mathbf{Y}$ que é explicada pelas covariáveis $\mathbf{X}$, e é expresso por:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{7}$$

Como o R^2 não é penalizado pelo número de covariáveis no modelo, basta acrescentarmos mais covariáveis para obter um R^2 maior. Assim, o mais correto é utilizar o **Coeficiente de Determinação Ajustado** pelos graus de liberdade de cada soma dos quadrados. Sua fórmula é dada por:

$$\begin{aligned} R^2_{adj} &= 1 - \frac{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-p}}{\frac{\sum_{i=1}^{n}(y_i - \bar{y})^2}{n-1}} \\ R^2_{adj} &= 1 - \frac{MSE}{MSTO} \end{aligned} \tag{8}$$

Ao contrário do R^2, o R^2_{adj} não é necessariamente crescente em $\mathbf{X}$, podendo inclusive reduzir com a inclusão de uma covariável para a qual sua contribuição marginal (ou sua soma dos quadrados extra) é não significativa.

2 Estimando Regressão Linear no R

Estimar regressão linear no R é bastante simples. Vamos utilizar um banco de dados `auto.RData`. O banco de dados `auto.RData` apresenta um estudo sobre 74 veículos em 1978 no mercado americano. Abaixo encontram-se descritas as variáveis contidas no banco de dados e seus respectivos rótulos:

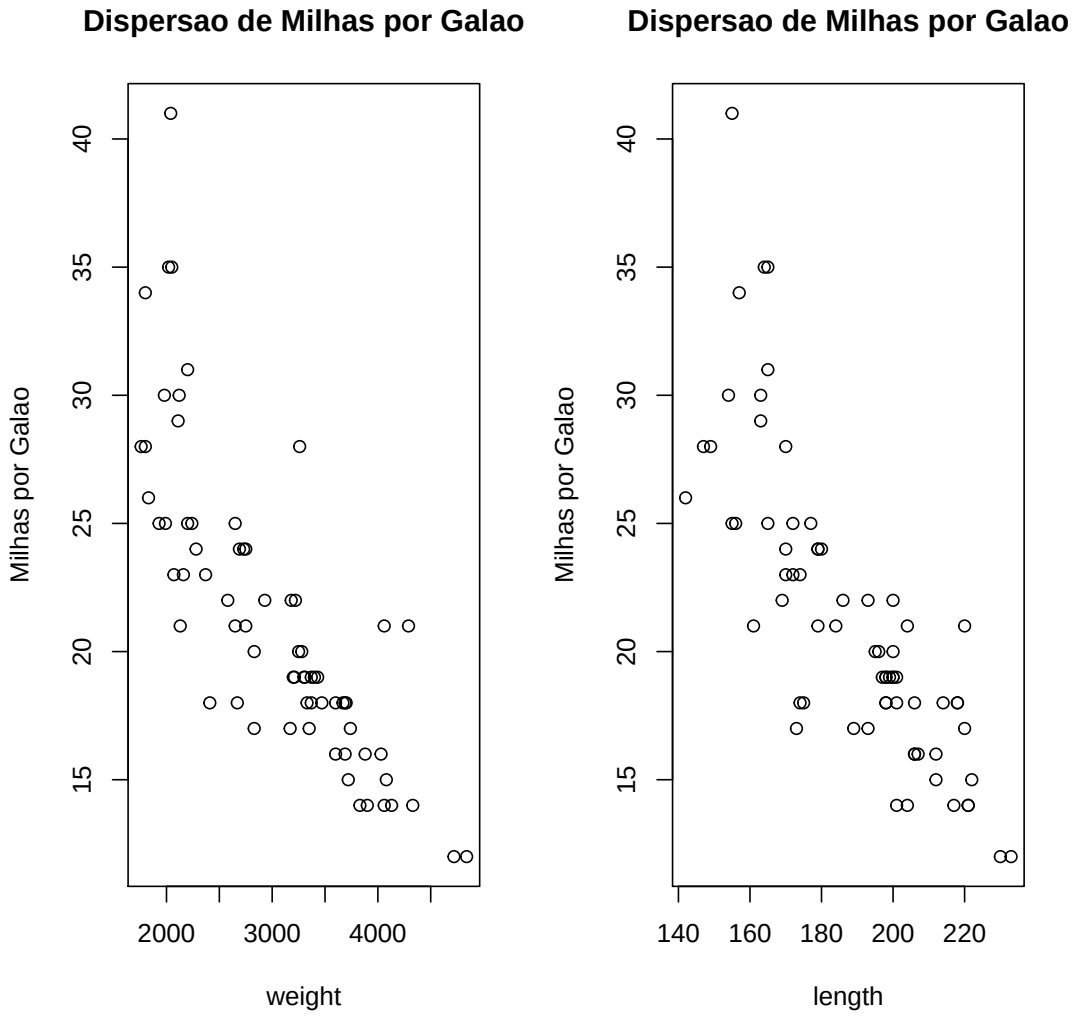
- `Make`: marca e modelo
- `Mpg`: milhas por galão (mpg)
- `Rep78`: número de reparos (contagem)
- `Headroom`: altura para cabeą (polegadas)
- `Trunk`: volume do porta-malas (pés cúbicos)
- `Weight`: peso (libras)
- `Length`: comprimento (polegadas)
- `Turn`: raio da menor curva-U feita pelo carro (pés)
- `Displacement`: volume trocado pelos pistõns em cada ciclo (pés cúbicos)
- `Gear ratio`: taxa de transmissão
- `Foreign`: variável binaria, 1 se o carro é estrangeiro, 0 caso contrário

No R, o modelo de regressão linear múltipla é ajustado através da função `lm()`, conforme exemplo a seguir. Vamos começar preparando a nossa área de trabalho, definindo o caminho do diretório, carregando o banco de dados e eliminando os casos de informações faltantes (`missing`).

```r
# Removendo objetos na area de trabalho
rm(list=ls(all=TRUE))

# Definindo o diretorio de trabalho
setwd("/Users/grguedes/APOSTILA/Modulo 2/")

# Verificando se o diretorio de trabalho mudou
getwd()

## [1] "/Users/grguedes/APOSTILA/Modulo 2"

# Carregando o pacote para importar dados do Stata
require(foreign)

## Loading required package:  foreign

# Criando um banco de dados sem missing
auto <- na.omit(read.dta("auto.dta"))
```

Agora vamos começar analisando a relação entre variáveis explicativas e dependente através de gráficos de dispersão. Lembre-se que esse tipo de gráfico só é apropriado para variáveis contínuas.

```r
# Definindo 2 graficos em mesma figura
par(mfrow=c(1,2))

# Fazendo um loop para dispersao entre Y e X
for(i in c(7,8)){

  plot(auto[,i],
       auto$mpg,
       ylab="Milhas por Galao",
       xlab=names(auto)[i],
       main="Dispersao de Milhas por Galao")

}
```



Finalmente, podemos passar para a análise de regressão. Vamos fazer um modelo da seguinte forma:

$$\begin{aligned} &\texttt{Modelo Populacional} \\ &mpg_i = \beta_0 + \beta_1 rep78_i + \beta_2 weight_i + \beta_3 length_i + \beta_4 foreign + \epsilon_i \\ &\texttt{Modelo Amostral} \\ &mpg_i = \hat{\beta}_0 + \hat{\beta}_1 rep78_i + \hat{\beta}_2 weight_i + \hat{\beta}_3 length_i + \hat{\beta}_4 foreign_i + \hat{\epsilon}_i \end{aligned} \tag{9}$$

Nesse modelo, nosso valor predito será dado por:

$$\hat{mpg}_i = \hat{\beta}_0 + \hat{\beta}_1 rep78_i + \hat{\beta}_2 weight_i + \hat{\beta}_3 length_i + \hat{\beta}_4 foreign_i \tag{10}$$

```r
# Rodando o modelo de regressao
fit1 <- lm(mpg~factor(rep78)+weight+length+foreign,data=auto)

# Apresentando a analise de variancia
anova(fit1)
```

```
## Analysis of Variance Table
##
## Response: mpg
##               Df  Sum Sq Mean Sq F value    Pr(>F)
## factor(rep78)  4  549.42  137.35 12.6560 1.469e-07 ***
## weight         1 1022.47 1022.47 94.2126 5.506e-14 ***
## length         1   45.55   45.55  4.1973   0.04480 *
## foreign        1   60.74   60.74  5.5965   0.02119 *
## Residuals     61  662.02   10.85
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Apresentando os coeficientes e medidas de ajuste
summary(fit1)
```

```
##
## Call:
## lm(formula = mpg ~ factor(rep78) + weight + length + foreign,
##     data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.0766 -1.3684 -0.1015  0.9421 11.5021
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    53.340930   6.653825   8.017 4.12e-11 ***
## factor(rep78)2  0.143993   2.617086   0.055   0.9563
## factor(rep78)3 -0.061010   2.416890  -0.025   0.9799
## factor(rep78)4  1.027410   2.529361   0.406   0.6860
## factor(rep78)5  4.305565   2.697729   1.596   0.1157
## weight         -0.002915   0.001684  -1.732   0.0884 .
## length         -0.123297   0.056466  -2.184   0.0329 *
## foreignForeign -3.090050   1.306192  -2.366   0.0212 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.294 on 61 degrees of freedom
## Multiple R-squared:  0.7171,Adjusted R-squared:  0.6846
## F-statistic: 22.09 on 7 and 61 DF,  p-value: 1.501e-14
```

```
# Calculando o valor predito
y_hat <- fit1$fitted.values
```

3 Regressão Linear Polinomial

Uma regressão é chamada de polinomial se é da forma:

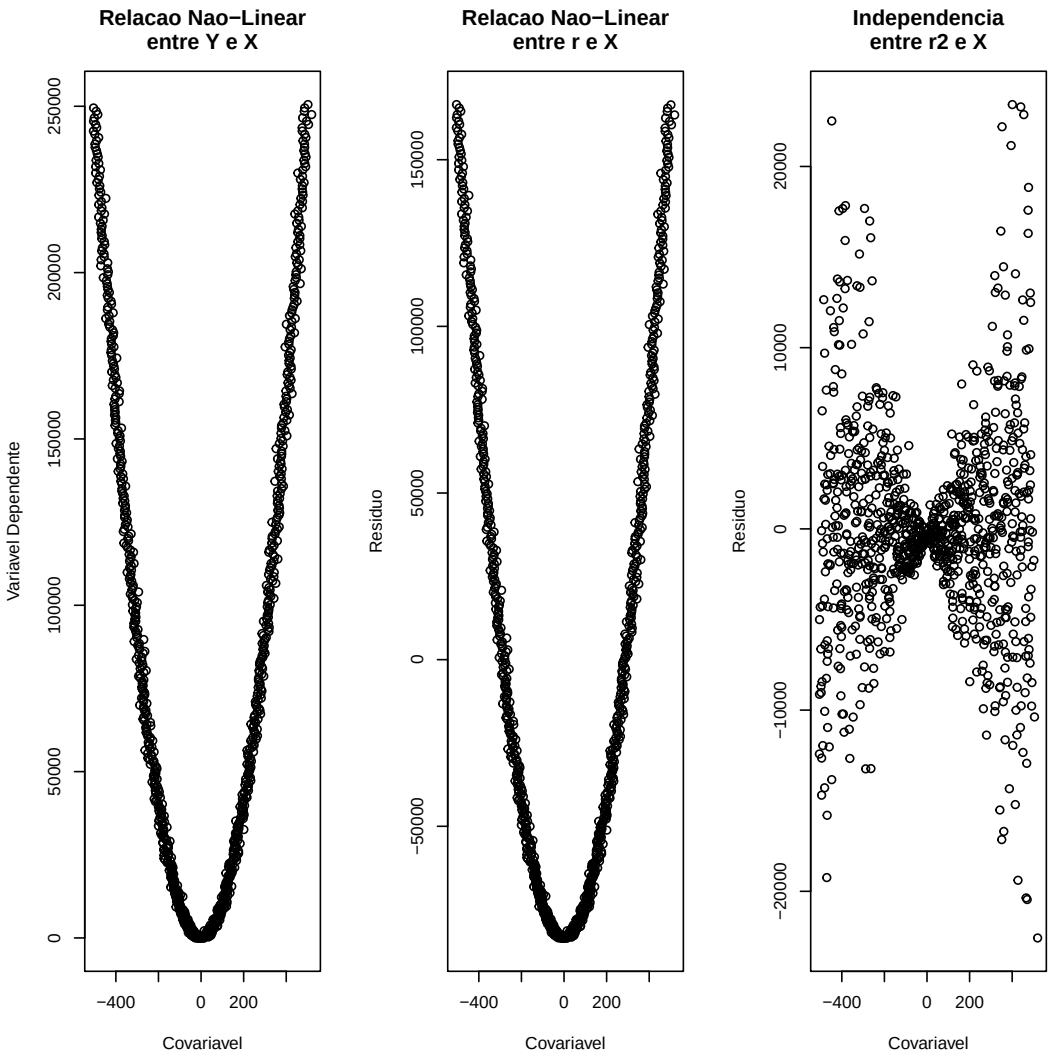
$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \ldots + \beta_p x_i^p + \epsilon_i \tag{11}$$

em que o p-ésimo x representa a variável original, x, elevada à potência p.

A regressão polinomial é um recurso importante para aproximar relações não-lineares entre Y e X em regressão linear múltipla. O caso abaixo apresenta uma relação não linear entre Y e X que pode ser facilmente aproximada linearmente a partir do uso das regressões polinomiais. A relação entre elas é da forma:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \epsilon_i \tag{12}$$

```
# Gerando a funcao polinomial estocastica
x<-c(-500:500)
y<-x+x^2
yvar<-y+rnorm(length(y),0,5)
xvar <- x+rnorm(length(x),0,10)

# Regressao sem termo polinomial
fit<-lm(yvar~xvar)
r <- fit$residuals

# Gerando termo quadratico
x2var <-xvar*xvar

# Regressao com termo polinomial
fit2<-lm(yvar~xvar+x2var)
r2 <- fit2$residuals

# Comparando os residuos
par(mfrow=c(1,3))
plot(xvar,yvar,
     ylab="Variavel Dependente",
     xlab="Covariavel",
     main="Relacao Nao-Linear\nentre Y e X")

plot(xvar,r,
     ylab="Residuo",
     xlab="Covariavel",
     main="Relacao Nao-Linear\nentre r e X")

plot(xvar,r2,
     ylab="Residuo",
     xlab="Covariavel",
     main="Independencia\nentre r2 e X")
```



Veja que a relação entre Y e X é claramente quadrática, e caso utilizemos uma regressão linear entre Y e X estaremos cometendo um erro de aproximação funcional considerável. Isso fica evidente quando comparamos o resíduo do modelo linear com X. A solução foi incluir o termo quadrático na regressão. Veja que o último gráfico tem os valores do resíduo contra X, eliminado o efeito do termo quadrático, dispersos em torno de 0. É o que queremos. Voltaremos mais a esse ponto na sessão de análise de resíduos. Por enquanto vamos ignorar a forma de gravata borboleta desse resíduo.

As regressões polinomiais apresentam alguns problemas típicos:

1. intercolinearidade sempre presentes.

2. modelo polinomial de ordem $n-1$ numa amostra de tamanho N sempre possuirá aderência perfeita aos dados.

3. extrapolação e mesmo interpolação ficam altamente instáveis em modelos polinomiais (previsão e resultados ilógicos).

4. polinômios complexos podem ser difíceis de interpretar.

Como visto no item (1) acima, as regressões polinomiais induzem intercolinearidade entre os monômios $(X, X^2, \ldots, X^p)$. A intercolinearidade é um problema nas estimações em regressão linear, pois afeta tanto os valores de β quanto $Var(\beta)$. Em algebra matricial,

uma matriz quadrada simétrica só admite inversa caso não haja dependência linear entre suas linhas ou colunas, o que nos leva ao pressuposto feito de **posto completo** da matriz **X**.

Quando há dependência linear entre linhas ou colunas de uma matriz, a determinante dessa matriz será nula. Como a inversa dessa matriz utiliza a determinante como denominador para cada elemento, os valores dariam ∞, impossibilitando que a matriz admita inversa. A intercolinearidade é um caso intermediário, em que a dependência linear não é perfeita, mas alta. Podemos pensar na intercolinearidade como o R^2 de uma coluna contra as demais na matriz **X**. Essa inclusive é a intuição por detrás da medida de inflação de variância - VIF (*variance inflation factor*).

Quanto maior a intercolinearidade, mais próximo a determinante da matriz será de zero, levando a grandes erros de arredondamento, causando viés nos estimadores amostrais, $\hat{\beta}$ (Equação 4) e $\hat{Var}(\beta)$ (Equação 6). A intercolinearidade causa uma série de problemas quando presente:

1. novas covariadas (acrescentadas ou excluídas) causam alteração nos coeficientes da regressão
2. Soma dos quadrados extras é afetada para uma covariada quando a colinear está no modelo
3. O desvio-padrão dos coeficientes é inflado (não-significância $\rightarrow$ redução do poder do `teste t`)
4. A não-significância de coeficientes colineares camufla uma possível associação linear entre X e Y

Existem dois tipos de diagnósticos para intercolinearidade:

1. Informal
    - mudanças acentuadas em $\hat{\beta}_j$ quando $X_{l \neq j}$ é acrescida ou excluída
    - Não-significância de $\hat{\beta}_j$ individuais para variáveis-estado
    - Efeito de $\hat{\beta}_j$ com sinal contrário ao teoricamente previsto
    - Elevada correlação pareada linear (matriz de correlação)
    - Amplos intervalos de confiança para $\hat{\beta}_j$ de variáveis-estado
2. Formal: Variance Inflaction Factor (VIF)

Algumas características do VIF:

- O fator de inflação de variância (VIF) nos dá uma métrica do impacto da intercolinearidade sobre nossos estimadores amostrais.
- Identifica ausência de colinearidade em situações sugeridas como colineares pelas evidências informais
- É baseado na comparação da matriz de variância/covariância num modelo normal relativo a um modelo com transformação de correlação (variáveis centralizadas e padronizadas)

- VIF individual (VIF_k) fornece uma medida da diferença entre o coeficiente padronizado estimado ($\hat{\beta}_k^{std}$) e o verdadeiro coeficiente padronizado (β_k^{std}).
- A média dos VIF_k revela o quanto a soma dos quadrados dos resíduos (SSE) seria aumentada em razão da intercolinearidade presente entre as covariadas.

As fórmulas para obtenção do VIF_k e do $\bar{VIF}$ encontram-se a seguir:

$$\begin{aligned} &\texttt{VIF Individual} \\ &VIF_k = (1 - R_k^2)^{-}1 \\ \\ &\texttt{VIF Médio} \\ &\bar{VIF} = \frac{\sum_{k=1}^{p-1}(1-R_k^2)^{-1}}{p-1} \end{aligned} \tag{13}$$

sendo R_k^2 o coeficiente de determinação múltipla quando X_k é regredido sobre $p-2$ outras covariadas X no modelo.

O critério para valores críticos não é consensual, mas devemos nos preocupar quando:

$$\begin{cases} VIF_k > 10 - 20\ 30 \\ \bar{VIF} > 1 \end{cases} \tag{14}$$

Vejamos agora um exemplo de uma regressão polinomial com nosso banco de dados de automóveis. Veja que o gráfico de dispersão entre as variáveis `mpg` e `weight` sugere uma relação não-linear que pode ser aproximada pelo modelo polinomial[1]. Utilizaremos um modelo do tipo:

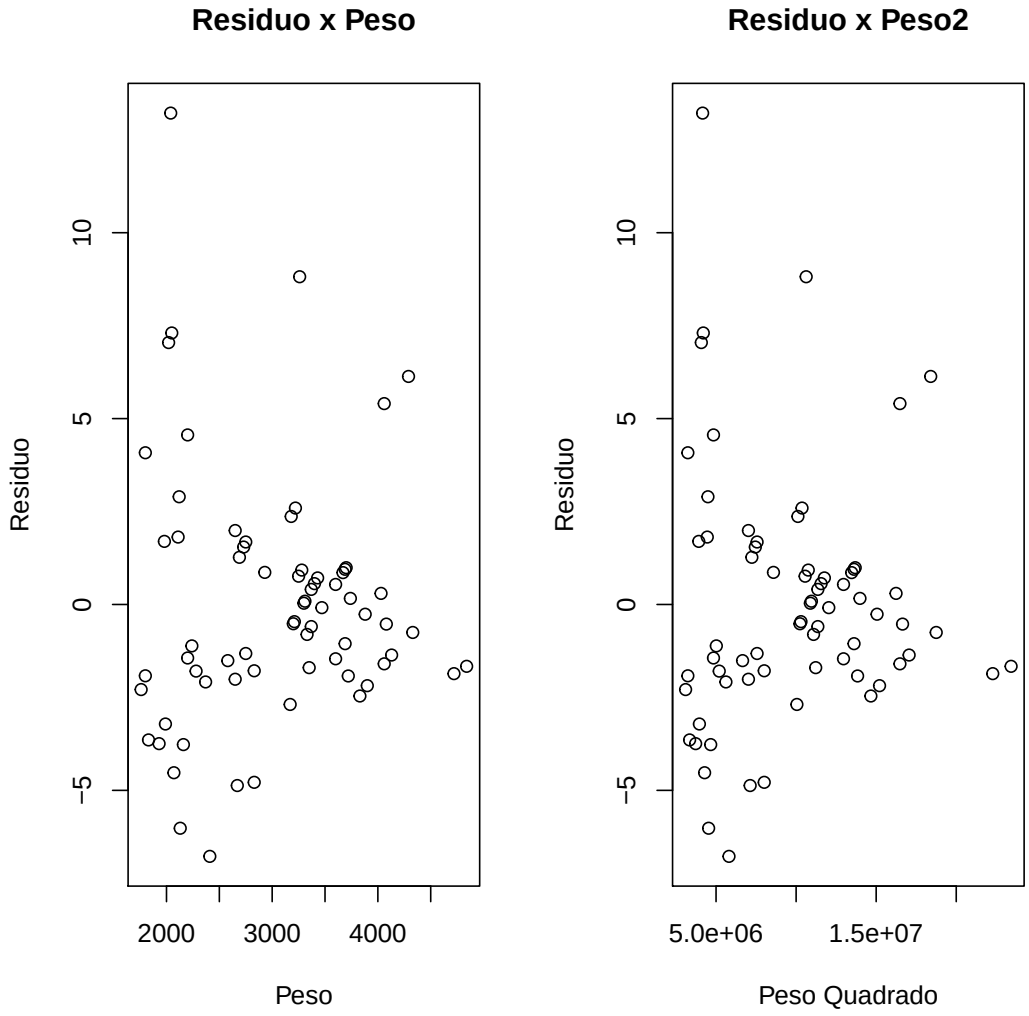
$$mpg_i = \hat{\beta}_0 + \hat{\beta}_1 weight_i + \hat{\beta}_2 weight_i^2 + \hat{\epsilon}_i \tag{15}$$

Veremos que o modelo da Equação (15) apresenta claramente um problema de alta intercolinearidade entre `weight` e `weight`^{2}.

```
# Criando a variavel peso ao quadrado
auto$weight2 <- auto$weight^2

# Ajustando o modelo:
fit10 <- lm(mpg~weight+weight2,data=auto)

# Pacote car e necessario para calcular o vif
require(car)

## Loading required package:  car

# Valor alto indica intercolinearidade
vif(fit10)

##   weight  weight2
## 58.77703 58.77703
```

[1]Para aproximações lineares mais gerais, podemos utilizar a aproximação por uma série de Taylor. Esse assunto será tratado na parte de modelos não-lineares para cálculo de variância das probabilidades preditas.

```r
# Verificando o residuo
res10 <- fit10$residuals
par(mfrow=c(1,2))
with(auto,
     plot(weight,res10,
     xlab="Peso",
     ylab="Residuo",
     main="Residuo x Peso"))
with(auto,
     plot(weight2,res10,
     xlab="Peso Quadrado",
     ylab="Residuo",
     main="Residuo x Peso2"))
```



Veja que o $VIF_k = 58.8$ para as duas variáveis, o que é bastante elevado. Uma solução simples para reduzir a intercolinearidade de modelos polinomiais é executar a `centralização` das variáveis Y e X antes de utilizar os termos quadráticos. Veja como a seguir:

```r
# Criando as variaveis centralizadas
auto$weightc <- auto$weight-mean(auto$weight)
auto$weightc2 <- auto$weightc^2

# Ajustando o modelo com as variaveis centralizadas
fit11 <- lm(mpg~weightc+weightc2,data=auto)

# Comparando as anovas
anova(fit10)
```

```
## Analysis of Variance Table
##
## Response: mpg
##           Df  Sum Sq Mean Sq  F value  Pr(>F)
## weight     1 1518.47 1518.47 129.0547 < 2e-16 ***
## weight2    1   45.17   45.17   3.8392 0.05429 .
## Residuals 66  776.56   11.77
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(fit11)
```

```
## Analysis of Variance Table
##
## Response: mpg
##           Df  Sum Sq Mean Sq  F value  Pr(>F)
## weightc    1 1518.47 1518.47 129.0547 < 2e-16 ***
## weightc2   1   45.17   45.17   3.8392 0.05429 .
## Residuals 66  776.56   11.77
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Comparando os coeficientes
summary(fit10)
```

```
##
## Call:
## lm(formula = mpg ~ weight + weight2, data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7754 -1.8600 -0.5185  0.9901 13.2178
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.060e+01  5.967e+00   8.481 3.66e-12 ***
## weight      -1.377e-02  4.022e-03  -3.424  0.00106 **
## weight2      1.269e-06  6.477e-07   1.959  0.05429 .
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.43 on 66 degrees of freedom
## Multiple R-squared:  0.6682,Adjusted R-squared:  0.6581
## F-statistic: 66.45 on 2 and 66 DF,  p-value: < 2.2e-16

summary(fit11)

##
## Call:
## lm(formula = mpg ~ weightc + weightc2, data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7754 -1.8600 -0.5185  0.9901 13.2178
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.050e+01  5.758e-01  35.610   <2e-16 ***
## weightc     -6.078e-03  5.281e-04 -11.510   <2e-16 ***
## weightc2     1.269e-06  6.477e-07   1.959   0.0543 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.43 on 66 degrees of freedom
## Multiple R-squared:  0.6682,Adjusted R-squared:  0.6581
## F-statistic: 66.45 on 2 and 66 DF,  p-value: < 2.2e-16

# Comparando os VIF
vif(fit10)

##   weight  weight2
## 58.77703 58.77703

vif(fit11)

##  weightc weightc2
## 1.013161 1.013161
```

Veja que essa simples solução reduziu o VIF_k de 58.8 para 1.0!

4 Regressão Linear com Termos Interativos

Na sessão anterior havíamos trabalhado com um modelo polinomial. Mas é possível trabalhar com um modelo mais geral, com a inclusão de **termos interativos**.

Veja que um termo quadrático, como utilizado anteriormente, é um tipo de termo interativo da variável x com ela mesma. Assim, o modelo mais geral de termos interativos pode incluir tanto termos quadrático, cúbicos, quanto interação entre covariáveis distintas.

Modelos lineares com termo interativo referem-se àqueles não podem ser expressos na forma aditiva:

$$\begin{aligned}
&\texttt{Modelos Aditivos}\\
&E[Y] = f_1(X_1) + f_2(X_2) + \ldots + f_{p-1}(X_{p-1})\\
&E[Y] = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2 + \beta_3 X_2\\
&\text{sendo}\\
&f_1(X_1) = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2\\
&f_2(X_2) = \beta_3 X_2\\
&\texttt{Modelos Não-Aditivos}\\
&E[Y] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2
\end{aligned} \tag{16}$$

O termo interativo é chamado na literatura técnica de:

- termo interativo de **produto cruzado**
- termo interativo **linear-por-linear**
- termo interativo **bilinear**

Há três tipos principiais de termos interativos:

1. Entre covariáveis contínuas
2. Entre covariáveis contínuas e discretas
3. Entre covariáveis discretas

Iremos destacar os casos (1) e (2) para efeitos de ilustração.

4.1 Termos Interativos entre Covariáveis Contínuas

O **modelo geral**, expresso em termos de **superfície de resposta média**, é definido por:

$$E[Y] = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2 + \beta_3 X_1 X_2 + \ldots + \beta p - 1 X_{p-1} \tag{17}$$

Vamos agora ver as consequências da inclusão do termo interativo para a interpretação dos coeficientes.

Interpretação dos coeficientes:

$$\begin{aligned}
\beta_0 &\rightarrow \ \frac{\partial E[Y]}{\partial X_0} = \beta_0\\
\beta_1 &\rightarrow \ \frac{\partial E[Y]}{\partial X_1} = \beta_1 + \beta_3 X_2\\
\beta_2 &\rightarrow \ \frac{\partial E[Y]}{\partial X_2} = \beta_2 + \beta_3 X_1
\end{aligned} \tag{18}$$

Veja que β_0 ficou inalterado com a inclusão do termo interativo. No entanto, β_1 e β_2 já não podem ser interpretados como num modelo aditivo. Ou seja, agora, o efeito de X_1 sobre Y depende do nível de X_2, o mesmo valendo para o efeito de X_2 sobre Y.

Consideremos os seguintes modelos:

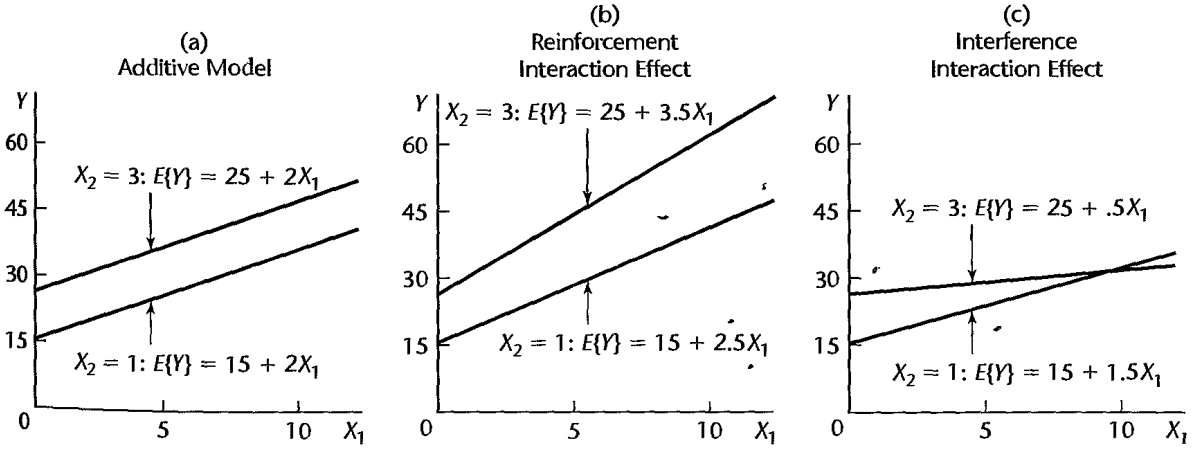
$$E[Y] = 10 + 2X_1 + 5X_2^2 \quad \texttt{Aditivo Mod. (1)}$$

$$E[Y] = 10 + 2X_1 + 5X_2^2 + 0.5X_1X_2 \quad \texttt{Interativo com Efeito Reforço Mod. (2)} \tag{19}$$

$$E[Y] = 10 + 2X_1 + 5X_2^2 - 0.5X_1X_2 \quad \texttt{Interativo com Efeito Interferência Mod. (3)}$$

Esses efeitos, sejam de reforço ou de interferência, podem ser visualizados na figura a seguir:

Fonte: Kutner et al. (2005) - Applied Linear Statistical Models



Vamos nos concentrar no efeito de X_1 sobre Y:

1. `Mod. (1):` $\frac{\partial E[Y]}{\partial X_1} = \beta_1 = 2$
2. `Mod. (2):` $\frac{\partial E[Y]}{\partial X_1} = \beta_1 + \beta_3 X_1 = 2 + 0.5X_2$
3. `Mod. (3):` $\frac{\partial E[Y]}{\partial X_1} = \beta_1 + \beta_3 X_1 = 2 - 0.5X_2$

Veja que no Mod. (1) o efeito de X_1 sobre Y independe do nível de X_2. Já nos Mod. (2) e (3), o efeito é diferente dependendo de para qual valor de X_2 estejamos olhando. No entanto, veja que no Mod. (3), o efeito de X_1 é:

$$\begin{cases} \texttt{Positivo} \rightarrow X_2 < 4 \\ \texttt{Nulo} \rightarrow X_2 = 4 \\ \texttt{Negativo} \rightarrow X_2 > 4 \end{cases} \tag{20}$$

Vejamos agora uma aplicação no R:

```r
# Ha duas formas de fazer interacao no R:
# criando a variavel de interacao antes
# usando o simbolo : na formula do lm()

# Modo 1: criando a variavel interativa
auto$weightalt<-with(auto,weight/1000)
auto$itwg <- with(auto,weightalt*gear_ratio)

# Regressao com Interacao
fit9 <- lm(mpg~weightalt+gear_ratio+
             itwg+factor(rep78)+foreign,data=auto)
summary(fit9)
```

```
##
## Call:
## lm(formula = mpg ~ weightalt + gear_ratio + itwg + factor(rep78) +
##     foreign, data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8089 -1.7532  0.0105  1.1472  9.2186
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -12.7940    12.9824  -0.985 0.328340
## weightalt        8.5975     3.8343   2.242 0.028655 *
## gear_ratio      17.7861     4.2635   4.172 9.89e-05 ***
## itwg            -4.9091     1.3278  -3.697 0.000474 ***
## factor(rep78)2  -0.6836     2.4196  -0.283 0.778508
## factor(rep78)3  -1.2691     2.2498  -0.564 0.574777
## factor(rep78)4  -0.6962     2.3532  -0.296 0.768368
## factor(rep78)5   3.8973     2.4968   1.561 0.123808
## foreignForeign  -6.2371     1.4615  -4.268 7.13e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.031 on 60 degrees of freedom
## Multiple R-squared:  0.7645,Adjusted R-squared:  0.7331
## F-statistic: 24.35 on 8 and 60 DF,  p-value: 3.616e-16
```

```r
# Modo 2: com o comando :
fit9 <- lm(mpg~weightalt+gear_ratio+weightalt:gear_ratio+
             factor(rep78)+foreign,data=auto)
summary(fit9)
```

```
##
## Call:
## lm(formula = mpg ~ weightalt + gear_ratio + weightalt:gear_ratio +
```

```
##     factor(rep78) + foreign, data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8089 -1.7532  0.0105  1.1472  9.2186
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -12.7940    12.9824  -0.985 0.328340
## weightalt              8.5975     3.8343   2.242 0.028655 *
## gear_ratio            17.7861     4.2635   4.172 9.89e-05 ***
## factor(rep78)2        -0.6836     2.4196  -0.283 0.778508
## factor(rep78)3        -1.2691     2.2498  -0.564 0.574777
## factor(rep78)4        -0.6962     2.3532  -0.296 0.768368
## factor(rep78)5         3.8973     2.4968   1.561 0.123808
## foreignForeign        -6.2371     1.4615  -4.268 7.13e-05 ***
## weightalt:gear_ratio  -4.9091     1.3278  -3.697 0.000474 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.031 on 60 degrees of freedom
## Multiple R-squared:  0.7645,Adjusted R-squared:  0.7331
## F-statistic: 24.35 on 8 and 60 DF,  p-value: 3.616e-16
```

Veja que neste exemplo temos um efeito de interferência, uma vez que o termo interativo tem um coeficiente negativo. Assim, poderiamos calcular o efeito do peso sobre o desempenho de combustível como:

$$\begin{cases} \frac{\partial E[mpg]}{\partial weight} = 8.5975 - 4.9091(gear\_ratio) \\ \\ \frac{\partial E[mpg]}{\partial weight|gear_{min}} = 8.5975 - 4.9091 * 2.190 = -3.15 \\ \\ \frac{\partial E[mpg]}{\partial weight|gear_{baixo}} = 8.5975 - 4.9091 * 2.544 = -3.89 \\ \\ \frac{\partial E[mpg]}{\partial weight|gear_{alto}} = 8.5975 - 4.9091 * 3.292 = -7.56 \\ \\ \frac{\partial E[mpg]}{\partial weight|gear_{max}} = 8.5975 - 4.9091 * 3.890 = -10.50 \end{cases} \tag{21}$$

4.2 Termos Interativos entre Covariáveis Contínuas e Categóricas

Diferentemente do modelo anterior, em que o β_0 não era alterado com a introdução do termo interativo, em modelos com variável binária o β_0 incorpora a categoria base dessa variável. Ou seja, caso não houvesse termo interativo, a equação seria representada por

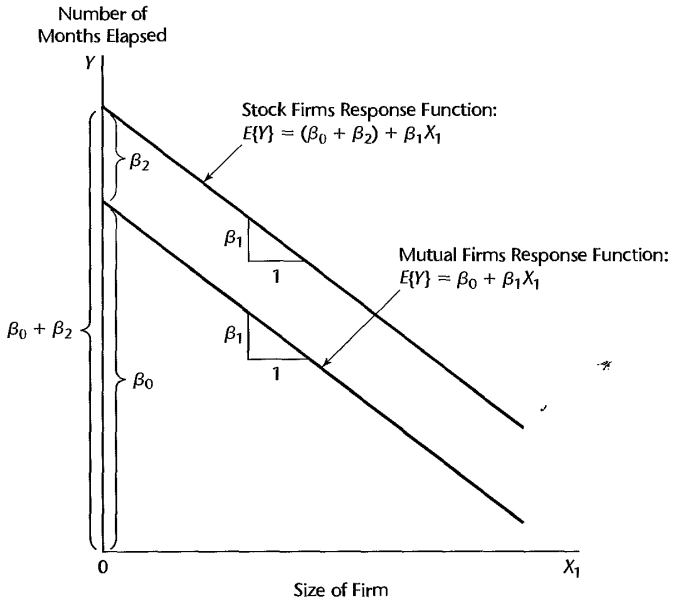
duas retas distintas, conforme abaixo:

$$\begin{cases} \texttt{Quando } D = 1 \\ E[Y] = \beta_0 + \beta_1 C + \beta_2 \\ E[Y] = (\beta_0 + \beta_2) + \beta_1 C \\ \\ \texttt{Quando } D = 0 \\ E[Y] = \beta_0 + \beta_1 C \end{cases} \tag{22}$$

Essas retas podem ser representadas graficamente como na Figura (4.2) abaixo:

Fonte: Kutner et al. (2005) - Applied Linear Statistical Models

FIGURE 8.11 Illustration of Meaning of Regression Coefficients for Regression Model (8.33) with Indicator Variable X_2—Insurance Innovation Example.



Agora, vamos entender como o modelo se comportaria se incluíssemos um termo interativo entre uma variável contínua e uma variável binária. Isso simplifica a interpretação e a visualização gráfica.

O **modelo geral**, expresso em termos de **superfície de resposta média**, é definido por:

$$E[Y] = \beta_0 + \beta_1 C + \beta_2 D + \beta_3 CD \tag{23}$$

em que C é a variável contínua e D a variável binária. Vamos agora ver as consequências da inclusão do termo interativo para a interpretação dos coeficientes.

Interpretação dos coeficientes:

$$\begin{aligned} \beta_0 &\rightarrow \frac{\partial E[Y]}{\partial X_0} = \beta_0 \\ \beta_1 &\rightarrow \frac{\partial E[Y]}{\partial C} = \beta_1 + \beta_3 D \\ \beta_2 &\rightarrow \frac{\partial E[Y]}{\partial D} = \beta_2 + \beta_3 C \end{aligned} \tag{24}$$

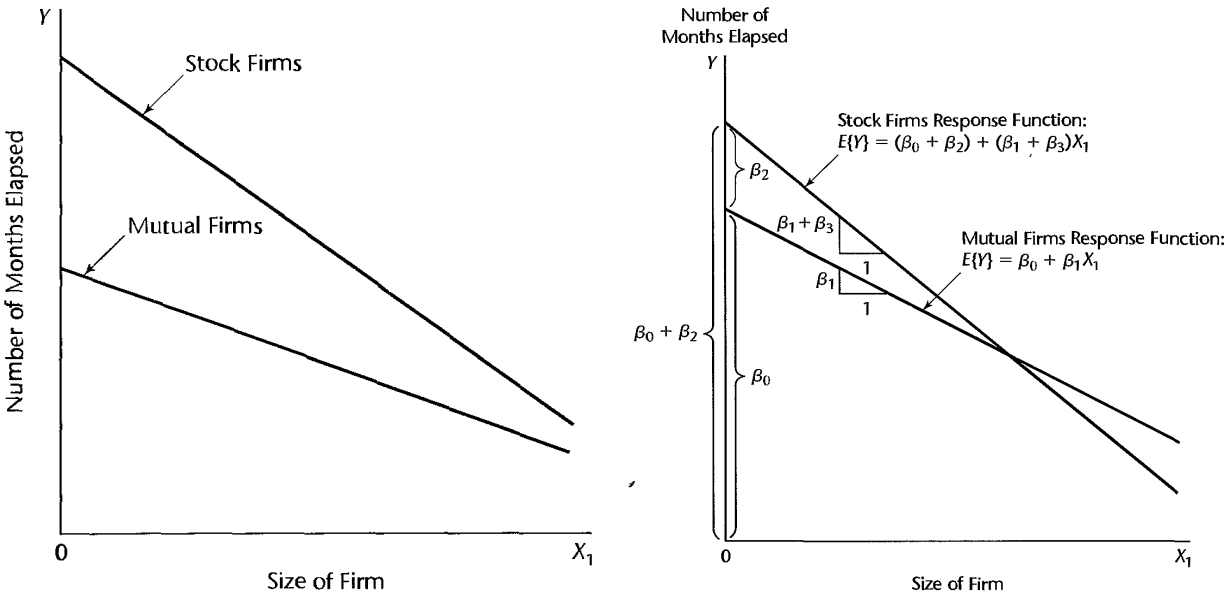
Como a variável binária assume apenas os valores 0 e 1, teríamos as seguintes equações:

$$\begin{aligned} &\texttt{Quando } D = 1 \\ &E[Y] = (\beta_0 + \beta_2) + (\beta_1 + \beta_3)C \end{aligned} \tag{25}$$

$$\begin{aligned} &\texttt{Quando } D = 0 \\ &E[Y] = \beta_0 + \beta_1 C \end{aligned}$$

Diz-se que há **interação ordinal** quando as funções de resposta não se cruzam dentro do escopo do modelo. Há **interação não-ordinal**, por outro lado, quando as funções de resposta se cruzam dentro do escopo do modelo. A figura abaixo mostra um exemplo de interação ordinal no Painel A e não-ordinal no Painel B:

Fonte: Kutner et al. (2005) - Applied Linear Statistical Models



Podemos calcular os efeitos, tanto da variável contínua quanto da categórica binária, com as fórmulas a seguir:

$$\begin{aligned} &\texttt{Efeito de C} \\ &\begin{cases} D = 0 \rightarrow \frac{\partial E[Y]}{\partial C} = \beta_1 \\ \\ D = 1 \rightarrow \frac{\partial E[Y]}{\partial C} = \beta_1 + \beta_3 \end{cases} \end{aligned} \tag{26}$$

$$\begin{aligned} &\texttt{Efeito de D} \\ &\begin{cases} C = \min C \rightarrow \frac{\partial E[Y]}{\partial D} = \beta_2 + \beta_3 \min C \\ \\ C = \max C \rightarrow \frac{\partial E[Y]}{\partial D} = \beta_2 + \beta_3 \max C \end{cases} \end{aligned}$$

Vejamos um exemplo prático no R.

```r
# Ha duas formas de fazer interacao no R:
# criando a variavel de interacao antes
# usando o simbolo : na formula do lm()

# Modo 1: criando a variavel interativa
auto$foreign <-as.numeric(auto$foreign)
auto$itwf <- with(auto,weightalt*foreign)

# Regressao com Interacao
fit8 <- lm(mpg~weightalt+foreign+
             itwf,data=auto)
summary(fit8)

##
## Call:
## lm(formula = mpg ~ weightalt + foreign + itwf, data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7187 -2.1520 -0.4724  0.7492 13.3141
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   29.884      6.877   4.346 4.98e-05 ***
## weightalt     -1.250      2.533  -0.494   0.6232
## foreign        9.863      5.377   1.834   0.0712 .
## itwf          -4.748      2.204  -2.154   0.0349 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.398 on 65 degrees of freedom
## Multiple R-squared:  0.6793,Adjusted R-squared:  0.6646
## F-statistic:  45.9 on 3 and 65 DF,  p-value: 4.77e-16

# Modo 2: com o comando :
fit8 <- lm(mpg~weightalt+foreign+weightalt:foreign,
           data=auto)
summary(fit8)

##
## Call:
## lm(formula = mpg ~ weightalt + foreign + weightalt:foreign, data = auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7187 -2.1520 -0.4724  0.7492 13.3141
##
## Coefficients:
```

```
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          29.884      6.877   4.346 4.98e-05 ***
## weightalt            -1.250      2.533  -0.494   0.6232
## foreign               9.863      5.377   1.834   0.0712 .
## weightalt:foreign    -4.748      2.204  -2.154   0.0349 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.398 on 65 degrees of freedom
## Multiple R-squared:  0.6793,Adjusted R-squared:  0.6646
## F-statistic:  45.9 on 3 and 65 DF,  p-value: 4.77e-16
```

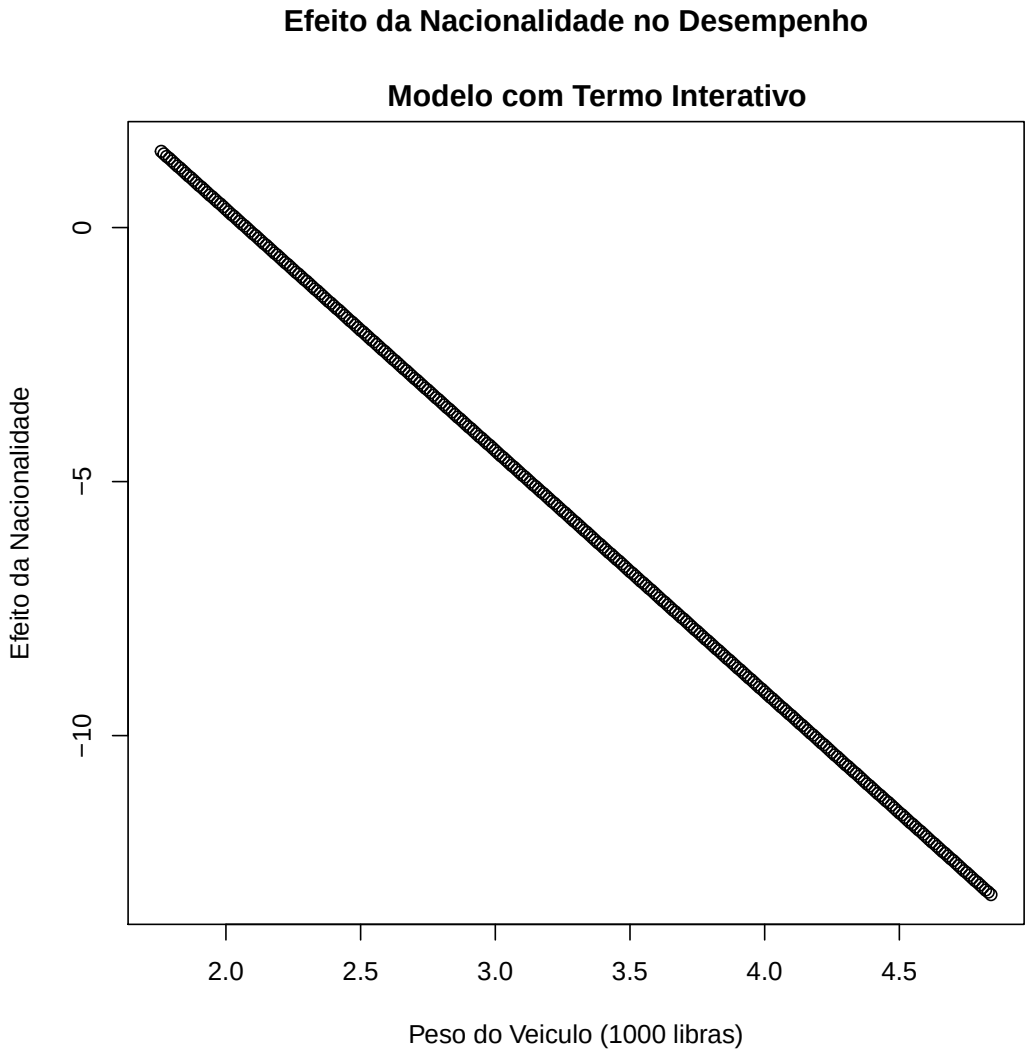
Veja que neste exemplo o termo interativo não foi significativo. Apenas para fins didáticos, no entanto, vamos ignorar a significância estatística e entender os efeitos. Assim, poderiamos calcular o efeito do peso e sobre o desempenho de combustível como:

$$\text{Efeito do Peso}$$
$$\begin{cases} Foreign = 0 \rightarrow \frac{\partial E[mpg]}{\partial weight} = -1.25 \\ \\ Foreign = 1 \rightarrow \frac{\partial E[mpg]}{\partial weight} = -1.25 - 4.75 = -6.0 \end{cases} \tag{27}$$

$$\text{Efeito da Nacionalidade}$$
$$\begin{cases} \min Peso \rightarrow \frac{\partial E[mpg]}{\partial Foreign} = 9.86 - 4.75 * 1.76 = 1.5 \\ \\ \max Peso \rightarrow \frac{\partial E[mpg]}{\partial Foreign} = 9.86 - 4.75 * 4.84 = -13.13 \end{cases}$$

Vejamos o gráfico do efeito dos carros importados sobre o desempenho de combustível:

```
efeito <- round(9.86-4.75*(seq(from=1.76, to=4.84,by=0.01)),digits=2)
peso <- seq(from=1.76, to=4.84,by=0.01)
par(mfrow=c(1,1))
plot(peso,efeito,
     ylab="Efeito da Nacionalidade",
     xlab="Peso do Veiculo (1000 libras)",
     main="Efeito da Nacionalidade no Desempenho\n
     Modelo com Termo Interativo")
```



Embora os termos interativos sejam uma forma atraente de incorporar não linearidade e condicionalidade de efeitos em regressões lineares, alguns cuidados devem ser tomados:

1. Em modelos com muitas covariáveis, a parametrização com termos interativos pode se tornar excessiva, reduzindo dramaticamente os graus de liberdade da soma dos quadrados dos resíduos. Isso é particularmente sério em amostras pequenas.

2. Uma alternativa é testar o resíduo gerado por um modelo aditivo contra todos os termos interativos. Selecionam-se os termos interativos que apresentarem algum padrão claro na análise de resíduos.

5 Análise de Resíduos

A análise de resíduos é uma das mais poderosas ferramentas para diagnóstico informal de qualidade de ajuste do modelo de regressão proposto.

Uma das primeiras funções atribuídas à análise de resíduo é verificar se estes estão distribuídos como:

$$\hat{\epsilon}_i \sim N(0, \sigma^2)$$

A normalidade dos resíduos é importante para manter a consistência do $\hat{\beta}$, embora o estimador da Equação (4) seja resistente a desvios pequenos da normalidade. Como será visto

adiante, a normalidade também pode ser aproximada com transformações relativamente simples.

A heterocedasticidade dos resíduos, no entanto, é a violação de pressupostos do modelo de regressão clássico que apresenta maiores problemas. O principal deles é o de que σ^2 não constante inviabiliza a construção do intervalo de confiança para o valor predito, uma vez que ele mudará em função do valor de X. Ou seja, a heterocedasticidae viola o pressuposto:

$$Var(\epsilon|\mathbf{x} = \sigma^2)$$

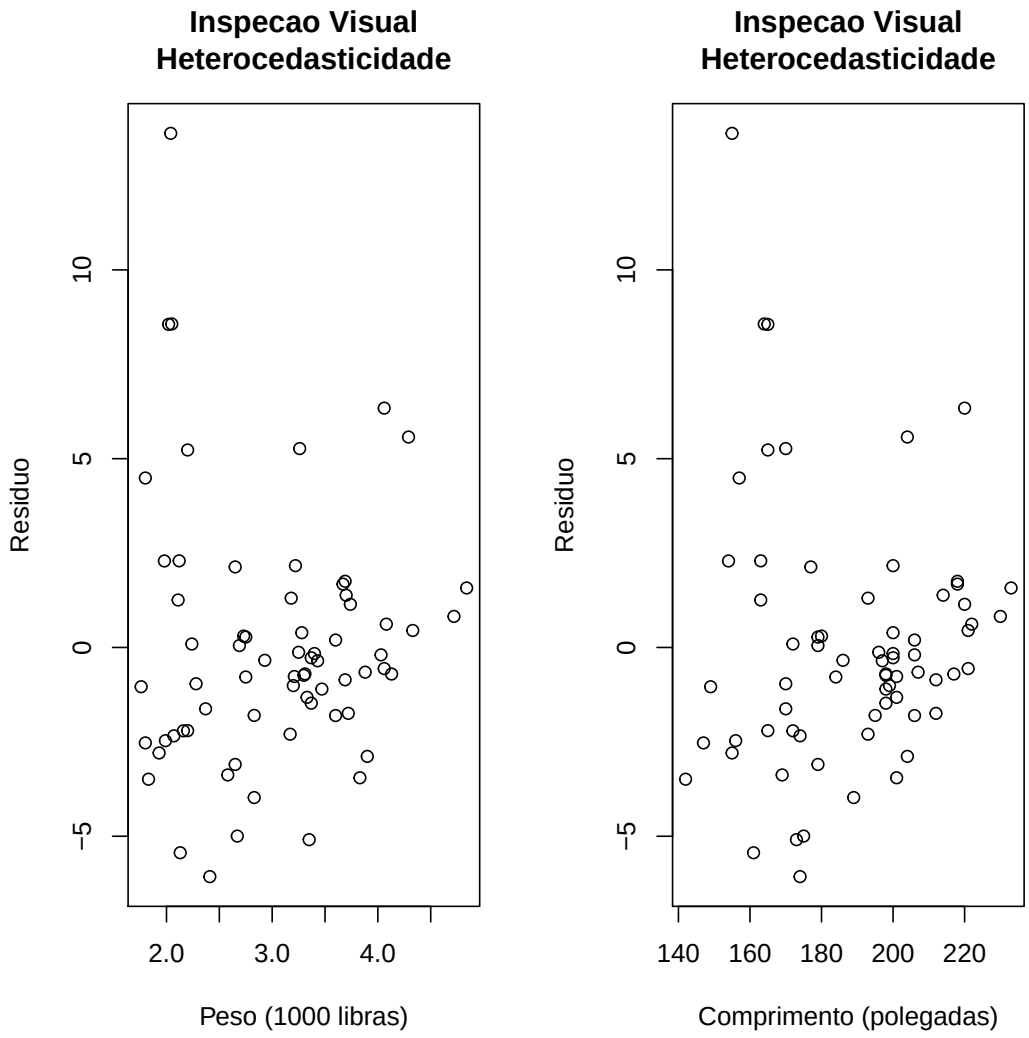
Existem duas formas para detectar heterocedasticidade dos resíduos:

1. **Visual**: através de gráficos de dispersão
2. **Formal**: através de testes estatísticos

5.1 Diagnóstico Visual de Heterocedasticidade

Se o resíduo fosse homocedástico, ele deveria ser distribuído com a mesma largura para todos os níveis de X. Para fazer o diagnóstico visual no R, siga os passos a seguir:

```
# Calculando o residuo
fit1 <- lm(mpg~weightalt+length+foreign,data=auto)
res <- fit1$residuals

par(mfrow=c(1,2))
# Analisando residuo contra covariavel do modelo

#leve<-with(auto,auto[weightalt<mean(weightalt),])
#pesado<-with(auto,auto[weightalt>=mean(weightalt),])

#curto<-with(auto,auto[length<mean(length),])
#longo<-with(auto,auto[length>=mean(length),])

auto$res <- NA
auto$res <-fit1$residuals

               plot(auto$weightalt,res,
               ylab="Residuo",
               xlab="Peso (1000 libras)",
               main="Inspecao Visual\nHeterocedasticidade")
#points(leve, col="red", pch=19)
#points(pesado, col="blue", pch=19)

plot(auto$length,auto$res,
     ylab="Residuo",
     xlab="Comprimento (polegadas)",
     main="Inspecao Visual\nHeterocedasticidade")
#points(curto, col="red", pch=19)
#points(longo, col="blue", pch=19)
```



Veja que em ambos os casos há uma tendência do resíduo apresentar maior dispersão para valores baixos das covariáveis `peso` e `comprimento`, um claro indício de heterocedasticidade.

5.2 Diagnósticos Formais de Heterocedasticidade

O próximo passo é fazer um teste formal para heterocedasticidade dos resíduos. Existem vários testes propostos na literatura, os quais variam em função de seus pressupostos quanto à **normalidade** e **condições assintóticas**. Por exemplo, os testes de `Bartlett` e `Levene` são robustos a desvios da normalidade, mas sensíveis a *outliers*. O teste não-paramétrico de `Fligner-Killeen`, por seu turno, é robusto a *outliers* e muito utilizado para testes em pequenas amostras, sendo um dos mais robustos testes de homocedasticidade para pequenos números. Se os dados são normais, o teste de `Bartlett` e `Breush-Pagan` são melhores.

Os quatro principais testes que veremos aqui são:

1. Breusch-Pagan / Cook-Weisberg
2. Fligner-Killeen
3. Levene

4. Multiplicador de Lagrange

5.2.1 Teste de Breusch-Pagan / Cook-Weisberg

O teste de Breusch-Pagan / Cook-Weisberg assume que os resíduos são independentes e normalmente distribuidos:

$$\hat{\epsilon}_i \text{ é i.i.d. e } \sim N(0, \sigma_i^2)$$

A sua equação de base é:

$$\sigma_i^2 = \gamma_0 + \gamma_1 X_{1i} + \ldots + \gamma_{p-1} X_{p-1,i} \tag{28}$$

O estimador amostral de σ_i^2 é dado por:

$$\hat{\sigma}_i^2 = (y_i - \hat{y}_i)^2 \tag{29}$$

A estrutura do teste é dada por:

$$\begin{cases} H_0 : \gamma = 0 \\ H_A : \gamma \neq 0 \end{cases} \tag{30}$$

A estatística de teste é:

$$X_{BP}^2 = \frac{SSR^*}{2} + \left(\frac{SSE}{n}\right)^2 \tag{31}$$

sendo:

$$SSR^* = \sum_{i=1}^{n} (\hat{\epsilon}_i^2 - \bar{\epsilon}^2)^2$$

e

$$\hat{\epsilon}_i^2 = \gamma_0 + \gamma_1 X_{1i} + \ldots + \gamma_{p-1} X_{p-1,i}$$

A estatística $X_{BP}^2 \sim \chi^2(1-\alpha, p-2)$.

É importante destacar que esse é um teste para grandes amostras e que assume normalidade dos resíduos. Já existem modificações no teste (utilizando a distribuição F ou t ao invés da χ^2, ou utilizando o Multiplicador de Lagrange) para relaxar o pressuposto de normalidade dos resíduos. Nesse caso, a equação de base torna-se:

$$ln(\sigma_i^2) = \gamma_0 + \gamma_1 X_{1i} + \ldots + \gamma_{p-1} X_{p-1,i} \tag{32}$$

A utilização de distribuições alternativas e modificações na equação base foram propostas por Koenker em 1981. O autor mostra que o teste de Breusch-Pagan possui tamanho assintótico incorreto, ou seja, incorreto nível nominal de significância, exceto no caso de resíduos estritamente Gaussianos. Suas proposições garantem que o teste possa ser utilizado, mesmo na presença de distúrbios não normais.

No R, o teste de Breusch-Pagan pode ser facilmente implementado com o *script* abaixo:

```
# Instalando e carregando o pacote
#install.packages("lmtest",dependencies=TRUE)
library(lmtest)
```

```
## Loading required package:  zoo
##
## Attaching package:  'zoo'
##
## The following objects are masked from 'package:base':
##
##    as.Date, as.Date.numeric

# Teste com pressuposto de Normalidade
bptest(mpg~weightalt+length+foreign,data=auto, studentize=FALSE)

##
##  Breusch-Pagan test
##
## data:  mpg ~ weightalt + length + foreign
## BP = 31.6033, df = 3, p-value = 6.344e-07

# Teste com correcao de Koenker
bptest(fit1, varformula=NULL, studentize=TRUE, data=auto)

##
##  studentized Breusch-Pagan test
##
## data:  fit1
## BP = 11.7439, df = 3, p-value = 0.008314

# Teste com correcao de Koenker
bptest(fit1, studentize = TRUE, data = auto)

##
##  studentized Breusch-Pagan test
##
## data:  fit1
## BP = 11.7439, df = 3, p-value = 0.008314
```

5.2.2 Teste de Fligner-Killeen

O teste de Fligner-Killeen é um teste de homogeneidade de variâncias que é robusto a desvios da normalidade. É um teste não paramétrico, portanto apropriado para pequenas amostras, e uma ótima alternativa ao teste de Bartlett quando há desvios de normalidade, assim como ao teste paramétrico de Levene quando se está trabalhando com pequenas amostras (Conover et al., 1981).

Basicamente, o teste de Fligner-Killeen é um método de posto linear simples para k amostras que usa os postos dos valores absolutos das amostras centralizadas, e pesos dados por:

$$a_i = qnorm\left(\frac{1}{2} + \frac{i}{2(n+1)}\right) \tag{33}$$

sendo `qnorm()` a função no R que retorna os quantis específicos da distribuição Normal.

A versão do teste disponível no R utiliza a centralização em torno da mediana de cada amostra.

A estrutura do teste de hipóteses é:

$$\begin{cases} H_0 : Var(X_{k=1}) = Var(X_{k=2}) = \ldots = Var(X_{k=K}) \\ H_A : \text{ Pelo menos um } k \text{ tem } Var(X_k) \neq Var(X_j) \end{cases} \tag{34}$$

A estatística de teste de Fligner-Killeen é definida como:

$$FK = \frac{\sum_{j=1}^{k} n_j(\bar{a}_j - \bar{a})^2}{s^2} \tag{35}$$

onde k é o número de grupos, n_j o tamanho do j-ésimo grupo, $\bar{a}_j$ é a média dos valores normalizados para o j-ésimo grupo, $\bar{a}$ é a média de todos os valores normalizados e s^2 é a variância de todos os valores normalizados. Veja que a obtenção dos valores normalizados é realizada com a aplicação da Equação (33).

A estatística FK segue uma distribuição qui-quadrado com os seguintes parâmetros:

$$FK \sim \chi^2(n, k-1)$$

Agora vamos implementar o teste no R:

```
# Regressao
fit2 <- lm(mpg~weightalt+factor(rep78)+foreign,data=auto)

# Implementando o teste
resfk <- fit2$residuals
fligner.test(resfk~auto$rep78)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  resfk by auto$rep78
## Fligner-Killeen:med chi-squared = 15.5951, df = 4, p-value =
## 0.003614

fligner.test(resfk~auto$foreign)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  resfk by auto$foreign
## Fligner-Killeen:med chi-squared = 7.0466, df = 1, p-value =
## 0.007941
```

5.2.3 Teste de Levene

O teste de Levene não depende do pressuposto de normalidade dos resíduos. No entanto, diferentemente do teste de Fligner-Killeen, requer amostras grandes o suficiente para tornar despresível a dependência dos termos do distúrbio amostral (estimados do mesmo $E[Y]$), com restrições:

- $\sum_{i=1}^{n} \hat{\epsilon}_i = 0$
- $Corr(x_s, \hat{\epsilon}_t) = 0$

Para a implementação do teste de Levene, devemos dividir a amostra em pelo menos dois grupos (baixos níveis de X e altos níveis de X), ou seja, $n = n_1 + n_2$, respectivamente.

A estatística de teste é dada por:

$$t_L^* = \frac{[n^{-1}(\epsilon_{i1} - \tilde{\epsilon}_1)] - [n^{-1}(\epsilon_{i2} - \tilde{\epsilon}_2)]}{s\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \tag{36}$$

sendo s o desvio-padrão das subamostras conjugadas e $\tilde{\epsilon}$ a mediana da subamostra.

A estatística de teste de Levene está distribuída segundo a distribuição t com:

$$t_L^* \sim t(1 - \alpha; n - 2)$$

O teste pode ser generalizado para mais de 2 grupos, como é rotineiramente implementado no R. Vamos agora dar um exemplo prático:

```
fit2 <- lm(mpg~weightalt+factor(rep78)+foreign,data=auto)

# Implementando o teste
library(car)
reslev <- fit2$residuals

# Teste com apenas 1 covariavel
auto$rep78 <- factor(auto$rep78)
leveneTest(reslev~auto$rep78)

## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value    Pr(>F)
## group  4  7.3532 6.154e-05 ***
##       64
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Teste com 2 ou + covariaveis
auto$foreign <- factor(auto$foreign)
leveneTest(reslev~auto$foreign*auto$rep78)

## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value    Pr(>F)
## group  7  4.9614 0.0001739 ***
##       61
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5.2.4 Teste do Multiplicador de Lagrange

Uma alternativa ao teste de Breusch-Pagan é o teste de Multiplicador de Lagrange. Esse teste relaxa o pressuposto da normalidade dos resíduos, assumindo apenas que esses são resíduos i.i.d. Para fazer o teste de ML para heterocedasticidade, deve-se primeiro ajustar o modelo de regressão múltipla e calcular o valor do resíduo. Em seguida, devemos estimar a equação auxiliar, definida como:

$$\hat{\epsilon}_i^2 = \gamma_0 + \gamma_1 X_{1i} + \ldots + \gamma_{p-1} X_{p-1,i} + \eta_i$$

A estatística de teste é definida como:

$$LM = nR_{MA}^2 \tag{37}$$

em que n é o tamanho da amostra e R_{MA}^2 corresponde ao coeficiente de determinação da equação auxiliar. Sob a hipótese H_0 de homocedasticidade, a estatística de teste LM é **assintoticamente distribuída** como:

$$LM \overset{a}{\sim} \chi^2(p-1)$$

No R, o teste é facilmente implementável através do seguinte *script*:

```
# Estimando a regressao linear
fit2 <- lm(mpg~weightalt+factor(rep78)+foreign,data=auto)

# Estimando a variancia do residuo
reslm2 <- fit2$residuals*fit2$residuals

# Estimando a equacao auxiliar
fit3 <- lm(reslm2~weightalt+factor(rep78)+foreign,data=auto)

# Estimando o teste
lmtest <- function(x,y) {
a <-summary(x)[["r.squared"]]*length(y)
dfree <- (x[["rank"]]-1)
pvalue <- 1-pchisq(a,dfree)
results <-cbind(round(a,digits=2),
                round(dfree,digits=0),
                round(pvalue,digits=4))
colnames(results) <- c("N","df","p-value")
return(results)
}

lmtest(fit3,reslm2)

##          N df p-value
## [1,] 21.86  6  0.0013
```

5.3 Análise Visual de Casos Influentes

A análise de resíduo também é útil para a detecção visual de casos influentes. Podemos definir 4 tipos diferentes de resíduos:

1. não padronizados
2. semi-studentizados
3. internamente studentizados
4. externamente studentizados

O resíduo normal é diretamente afetado pela escala de Y e, portanto, não é a melhor opção para identificação de casos extremos. A primeira alternativa é a utilização de uma padronização, conforme abaixo:

$$\texttt{Resíduo Semi-studentizado}$$

$$\epsilon_i^* = \frac{\epsilon_i}{\sqrt{MSE}}$$

$$\texttt{Casos Influentes}$$

$$\epsilon_i^* > |4| \tag{38}$$

Esse resíduo pode ser analisado como número de desvios-padrão em torno da média, $E[\epsilon_i] = 0$. Apesar de reduzir a influência da escala de Y, ele assume dois pressupostos bastante restritivos:

1. resíduos são homocedásticos
2. influência de cada observação amostral é idêntica

Como alternativa a esse resíduo, pode-se utilizar o **Resíduo Internamente Studentizado**, que pode ser calculado como:

$$\texttt{Resíduo Internamente Studentizado}$$

$$r_i^* = \frac{\epsilon_i}{\sqrt{s^2(\epsilon_i)}} = \frac{\epsilon_i}{\sqrt{MSE(1-h_{ii})}}$$

$$\texttt{Casos Influentes}$$

$$r_i^* > |4| \tag{39}$$

Veja que, diferentemente do resíduo semi-estudentizado, o resíduo internamente studentizado leva em consideração a influência de cada observação no cálculo da soma dos quadrados dos resíduos média, MSE; ou seja, esse resíduo é ponderado pela variância individual. No entanto, ele ainda considera o pressuposto de MSE constante.

Veja também que o termo $(1 - h_{ii})$ agora aparece no denominador da fórmula. Esse valor é exatamente a diagonal principal da **matriz chapéu**. A matriz chapéu é definida como:

$$\begin{gathered} \underset{n\times n}{\mathrm{H}} = \mathrm{X}(\mathrm{X}^{\mathrm{T}}\mathrm{X})^{-1}\mathrm{X}^{\mathrm{T}} \\ \hat{\mathrm{Y}} = \mathrm{HY} \\ \epsilon = (\mathrm{I} - \mathrm{H})\mathrm{Y} \\ \sigma^2(\epsilon) = \sigma^2((\mathrm{I} - \mathrm{H})) \end{gathered} \tag{40}$$

No caso de observações individuais, teríamos:

$$\begin{gathered} \sigma^2(\epsilon_i) = \sigma^2(1 - h_{ii}) \\ \sigma(\epsilon_i, \epsilon_j) = \sigma(0 - h_{ij}) = -h_{ij}\sigma^2 \ i \neq j \\ s^2(\epsilon_i) = MSE(1 - h_{ii}) \\ s(\epsilon_i, \epsilon_j) = -h_{ij}(MSE) \ i \neq j \end{gathered} \tag{41}$$

Como $0 \leq h_{ii} \leq 1$, quanto maior o valor de h_{ii} maior sua influência para o cálculo do MSE da regressão. Observações muito influentes apresentam $s^2(\epsilon_i)$ baixo, e elevam o MSE da regressão. Exatamente por isso, assumir um MSE constante ainda é uma limitação do resíduo internamente studentizado.

Uma alternativa a ele é o **Resíduo Externamente Studentizado**, o qual flexibiliza todos os pressustos restritivos: leva em consideração a influência da cada observação sobre o resíduo (ao incluir o h_{ii} na fórmula), além de levar em conta sua influência sobre o valor do MSE. Ele pode ser calculado a partir de:

$$\begin{gathered} \texttt{Resíduo Externamente Studentizado} \\ t_i^{=} \frac{\epsilon_i}{\sqrt{s^2_{(i)}(\epsilon_i)}} = \frac{\epsilon_i}{\sqrt{MSE_{(i)}(1-h_{ii})}} \\ \texttt{Análogo com 1 regressão} \\ t_i^{=} \epsilon_i \left[\frac{n-p-1}{SSE(1-h_{ii})-\epsilon^2} \right]^{\frac{1}{2}} \end{gathered} \tag{42}$$

O resíduo t_i segue uma distribuição de probabilidade de Bonferroni:

$$t_i \sim t(1 - \frac{\alpha}{2n}; n - p - 1)$$

Um caso é considerado extremo para Y caso:

$$t_i > t(1 - \frac{\alpha}{2n}; n - p - 1)$$

Como o teste de Bonferroni é bastante conservador, pode-se utilizar um nível de significância α=10%. Vejamos agora como estimar esses resíduos no R:

```r
# Estimando a regressao
fit <- lm(mpg~weightalt+factor(rep78)+foreign,data=auto)

# Residuo Comum
r <- fit$residuals

# Residuo Semi-studentizado
rss <- r/anova(fit)[["Mean Sq"]][4]

# Residuo Internamente Studentizado
ris  <- rstandard(fit)

# Residuo Externamente Studentizado
res <- rstudent(fit)

resTest <- function(x,alpha,y) {
 b<- qt((1-(alpha/length(y)*2)),
        (anova(x)[["Df"]][length(anova(x)[["Df"]])]-1))
 inf <- ifelse(y>b,1,0)
 results <- cbind(round(y,digits=2),round(b,digits=2),
                  round(inf,digits=0))
 colnames(results) <- c("Res Stud","Critic T", "Influent")
 return(results)
}

x<-data.frame(resTest(x=fit,alpha=0.05,y=res))
x[x$Influent==1,]

##    Res.Stud Critic.T Influent
## 71     4.36      3.1        1
```

6 Transformações de Variáveis

Na sessão anterior vimos que os resíduos da regressão podem apresentar três tipos de comportamento que comprometem os pressupostos básicos da regressão linear:

1. não-linearidade
2. heterocedasticidade
3. padrão sistemático

Uma característica importante é que resíduos heterocedásticos são, via de regra, resíduos não normais. Portanto, quando se faz algum tipo transformação nas variáveis que corrija a não linearidade da distribuição, também se acaba por diminiur a heterocedasticidade.

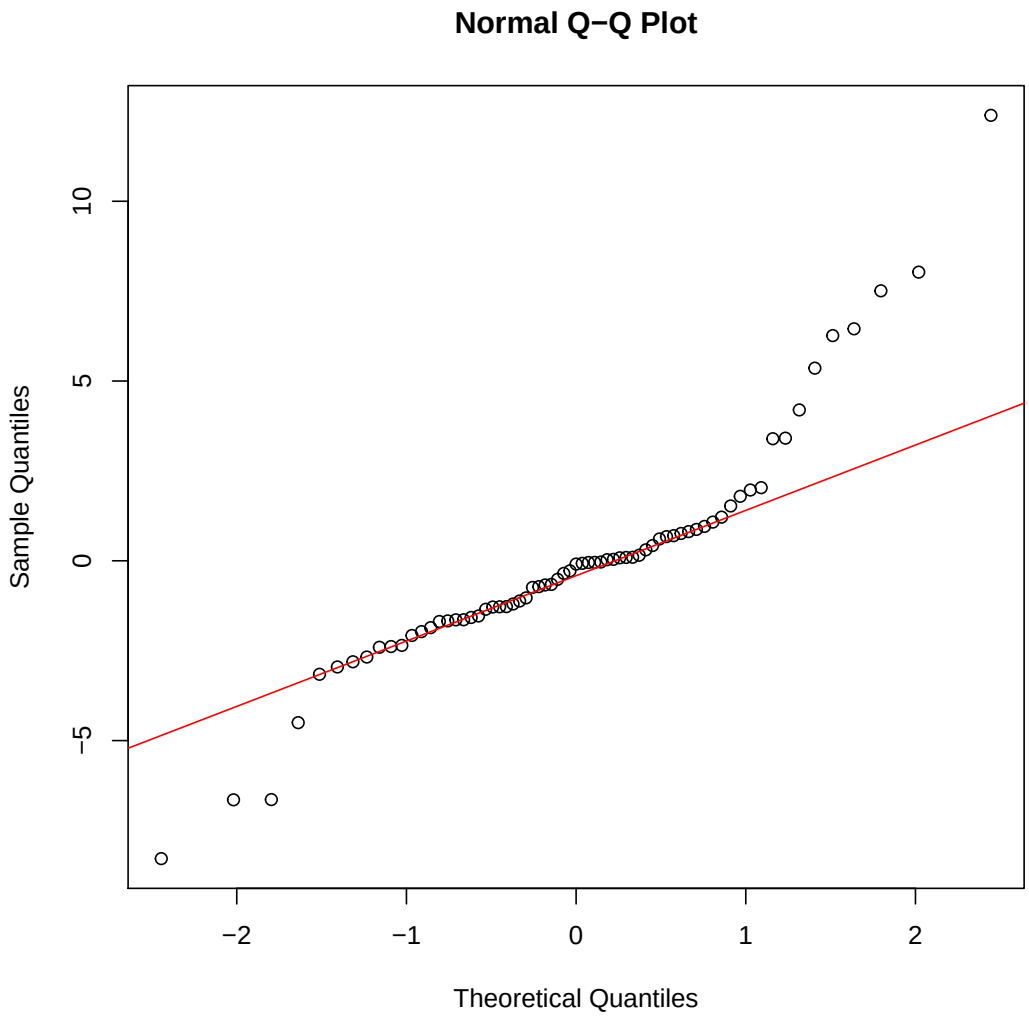
A primeira forma de verificar a normalidade dos resíduos é fazer um gráfico de distribuição conhecido como Gráfico de Quantis Normais. Caso os resíduos estivessem normalmente distribuídos, seus valores deveriam ser idênticos aos valores esperados sob condição de normalidade.

Os resíduos sob condição de normalidade podem ser calculados como:

$$E[\hat{\epsilon}_i] \sim N(0, \sigma^2)] = \sqrt{MSE}\left[z\left(\frac{k - 0.375}{n + 0.25}\right)\right]$$

sendo k o posto do resíduo. Vejamos como construir o gráfico de diagnóstico de desvio da normalidade no R.

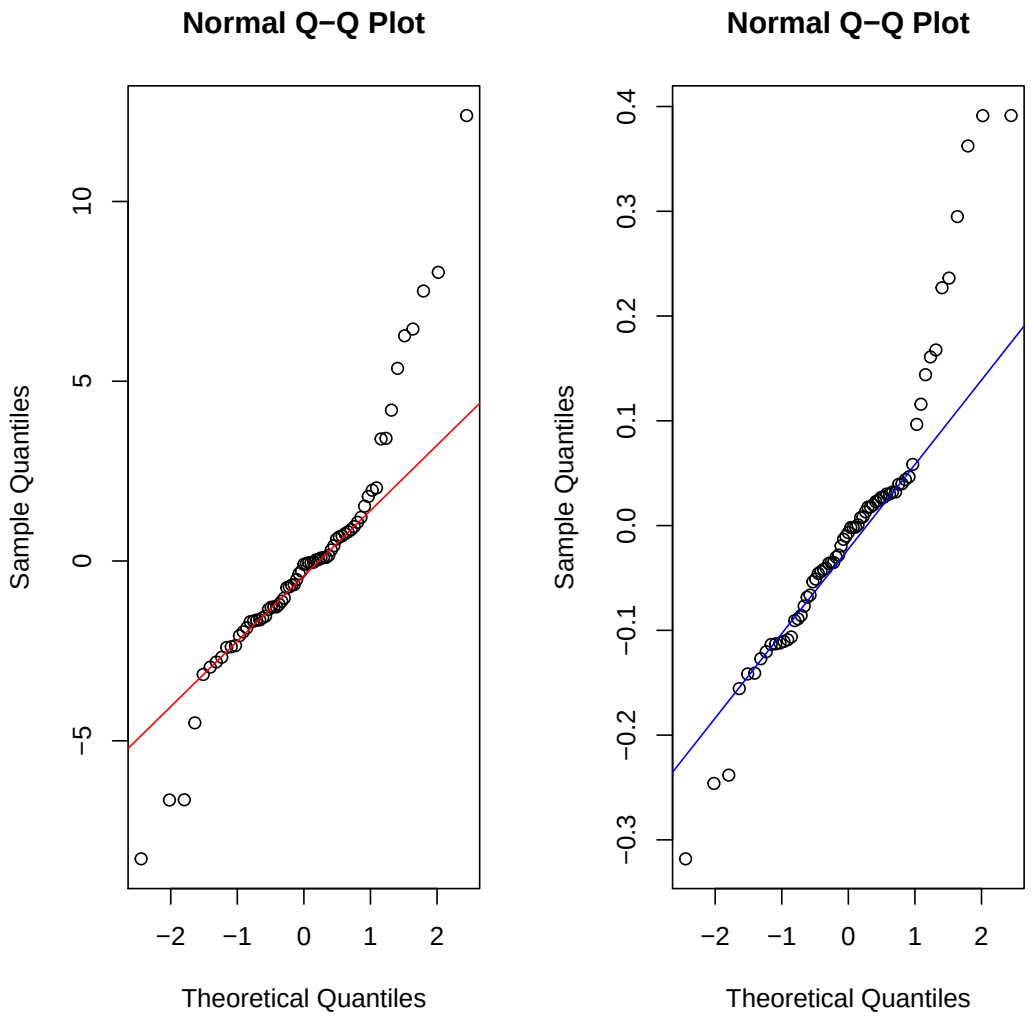
```
# Grafico de Quantil Normal
qqnorm(r)
qqline(r,col="red")
```



6.1 Transformações Logarítmicas

Veja que há um desvio substancial da normalidade. Caso efetuemos uma primeira transformação, tirando o logarítmo da variável dependente, pode ser que essa não-normalidade seja atenuada:

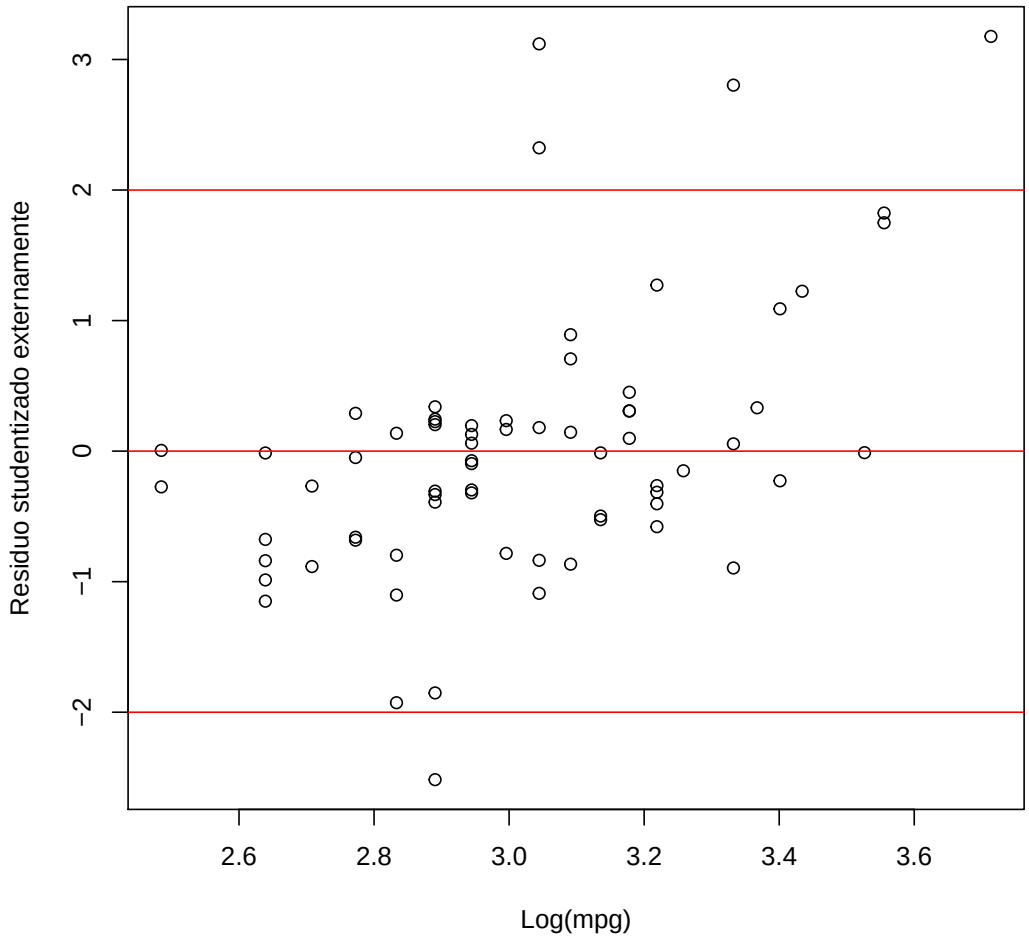
```
# Grafico de Quantil Normal
auto$lnmpg <- NA
auto$lnmpg <-with(auto,log(mpg))

fit <- lm(lnmpg~weightalt+factor(rep78)+foreign,data=auto)
r2 <- fit$residuals
par(mfrow=c(1,2))
qqnorm(r)
qqline(r,col="red")
qqnorm(r2)
qqline(r2,col="blue")
```



Vimos que a transformação logarítimica não pareceu suficiente para eliminar a não-linearidade dos resíduos. Podemos verificar esse resíduo sob outro ângulo, incluindo um teste de distribuição de Anderson-Darling. Veja a seguir:

```
res <- rstudent(fit)

# Inspecao visual
par(mfrow=c(1,1))
```

```r
plot(auto$lnmpg,res,
     ylab="Residuo studentizado externamente",
     xlab="Log(mpg)")
abline(h=c(-2,0,2),col="red")

# Teste de Anderson-Darling
require(nortest)

## Loading required package:  nortest

ad.test(res)

##
##  Anderson-Darling normality test
##
## data:  res
## A = 2.3421, p-value = 5.425e-06
```



A transformação logarítimica tem uma grande vantagem, que é sua facilidade de interpretação. A figura abaixo resume os 4 tipos possíveis de modelos com transformações

logarítmicas, com suas respectivas interpretações:

1. modelo lin-lin
2. modelo log-lin
3. modelo lin-log
4. modelo log-log

Summary of Functional Forms Involving Logarithms

Model	Dependent Variable	Independent Variable	Interpretation of β_1
level-level	y	x	$\Delta y = \beta_1 \Delta x$
level-log	y	$\log(x)$	$\Delta y = (\beta_1/100)\%\Delta x$
log-level	$\log(y)$	x	$\%\Delta y = (100\beta_1)\Delta x$
log-log	$\log(y)$	$\log(x)$	$\%\Delta y = \beta_1 \%\Delta x$

6.2 Transformação Box-Cox

Um recurso mais interessante, e mais geral, é a chamada transformação Box-Cox. Caso a hipótese de normalidade dos resíduos não seja satisfeita, deve-se considerar utilizar uma transformação na variável resposta ou em alguma variável explicativa. Ainda se podem inserir preditores que são funções de outros preditores ou termos interativos.

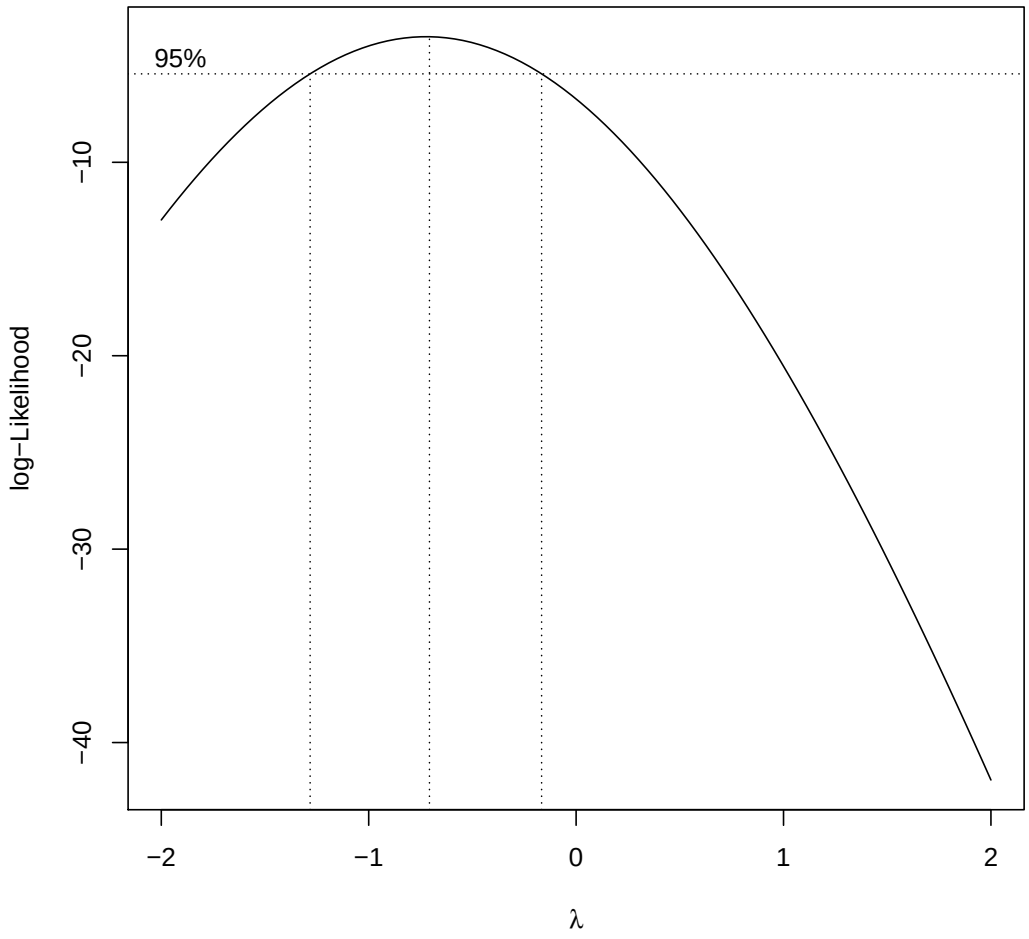
O método de Box-Cox é muito utilizado para determinar a transformação a ser utilizada. Fixado um λ, a transformação é dada por:

$$t_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda} \text{ se } \lambda \neq 0 \\ log(y) \text{ se } \lambda = 0 \end{cases} \tag{43}$$

Vejamos a implementação no R:

```
# Transformacao de box cox
require(MASS)

## Loading required package:  MASS

bc <- boxcox(fit1)

lambda <- bc$x[bc$y==max(bc$y)]
```

```r
tbc <- function(x,lambda){
  return((x^lambda-1)/lambda)
}

yt <- tbc(auto$price,lambda)

fit2 <- lm(yt~mpg+factor(rep78)+weightalt+foreign,data=auto)
```



O valor de λ é obtido por máxima verossimilhança. Uma limitação dessa transformação é que ela garante uma normalização da variável dependente, mas ignora a relação entre esta e as variáveis explicativas do modelo. Nesse sentido, podemos obter uma transformação que seja normalizada, mas que não seja a melhor transformação que expresse a relação linear entre Y e X.

Os modelos a seguir, também obtidos por máxima verossimilhança, são genéricos o suficiente para normalizar a variável dependente e assegurar uma melhor relação linear

com as variáveis transformadas. As equações serão:

$$
\begin{aligned}
&\texttt{Transformação de } Y \\
&\mathrm{y}^{\theta} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_{p-1} X_{p-1,i} + \epsilon_i \\
&\\
&\texttt{Transformação de } X \\
&\mathrm{y}{=}\beta_0 + \beta_1 X_1^{\lambda} + \beta_2 X_2^{\lambda} + \ldots + \beta_{p-1} X_{p-1,i}^{\lambda} + \epsilon_i \\
&\\
&\texttt{Transformação única de } X \texttt{ e } Y \\
&\mathrm{y}^{\lambda} = \beta_0 + \beta_1 X_1^{\lambda} + \beta_2 X_2^{\lambda} + \ldots + \beta_{p-1} X_{p-1,i}^{\lambda} + \epsilon_i \\
&\\
&\texttt{Transformação variável de } X \texttt{ e } Y \\
&\mathrm{y}^{\theta} = \beta_0 + \beta_1 X_1^{\lambda} + \beta_2 X_2^{\lambda} + \ldots + \beta_{p-1} X_{p-1,i}^{\lambda} + \epsilon_i
\end{aligned}
\tag{44}
$$