Análise de regressão linear múltipla para empresas de jornal

Gabriel D’assumpção de Carvalho

19/03/2024

Introdução

Este relatório tem como objetivo realizar uma análise exploratória seguida da proposição de um modelo de regressão linear múltipla para os dados contidos em *journals.txt*. O conjunto de dados compreende 180 empresas de jornais, cada uma caracterizada por 10 variáveis, descritas da seguinte forma:

1. title: Categórica
2. publisher: Categórica
3. society: Categórica
4. price: Numérica
5. pages: Numérica
6. charpp: Numérica
7. citations: Numérica
8. foundingyear: Numérica
9. subs: Numérica
10. field: Categórica

Análise Exploratória

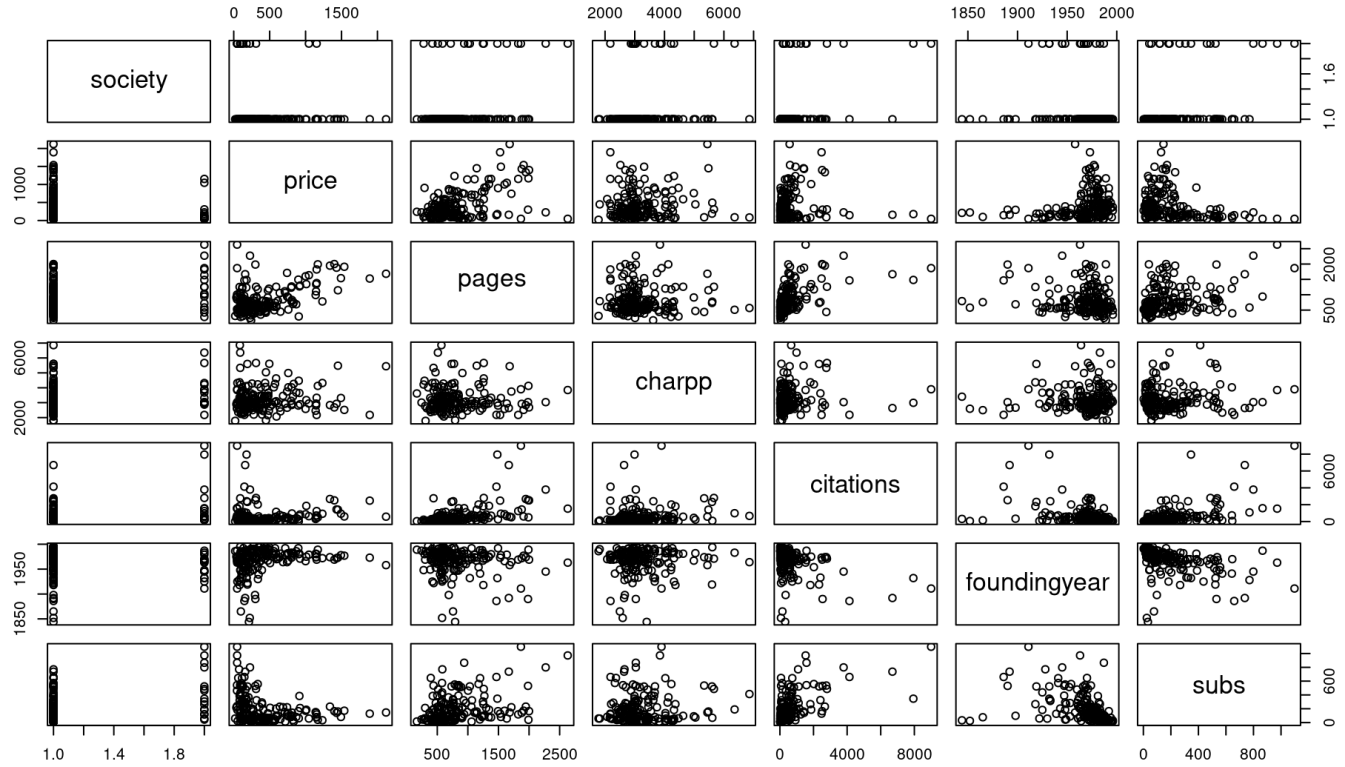
Neta etapa vamos fazer uma análise breve para verificar cada variável.

```
#Obtendo a tabela de dados do repositório
data <- read.table("https://raw.githubusercontent.com/gabrieldacarvalho/analise_regressao/main/multiple_linear_regression/journals/data_journals.txt")

summary(data)
```

```
##     title            publisher           society              price       
##  Length:180         Length:180         Length:180         Min.   :  20.0  
##  Class :character   Class :character   Class :character   1st Qu.: 134.5  
##  Mode  :character   Mode  :character   Mode  :character   Median : 282.0  
##                                                           Mean   : 417.7  
##                                                           3rd Qu.: 540.8  
##                                                           Max.   :2120.0  
##      pages             charpp       citations         foundingyear 
##  Min.   : 167.0   Min.   :1782   Min.   :  21.00   Min.   :1844  
##  1st Qu.: 548.8   1st Qu.:2715   1st Qu.:  97.75   1st Qu.:1963  
##  Median : 693.0   Median :3010   Median : 262.50   Median :1973  
##  Mean   : 827.7   Mean   :3233   Mean   : 647.06   Mean   :1967  
##  3rd Qu.: 974.2   3rd Qu.:3477   3rd Qu.: 656.00   3rd Qu.:1982  
##  Max.   :2632.0   Max.   :6859   Max.   :8999.00   Max.   :1996  
##       subs             field          
##  Min.   :   2.0   Length:180        
##  1st Qu.:  52.0   Class :character  
##  Median : 122.5   Mode  :character  
##  Mean   : 196.9                     
##  3rd Qu.: 268.2                     
##  Max.   :1098.0
```

Para analisar a relação entre variáveis, vamos nos concentrar nas variáveis society, price, pages, charpp, citations, foundingyear e subs. A variável society é categórica binária, enquanto as demais são numéricas. Isso nos permitirá avaliar tanto a associação entre a variável categórica e as variáveis numéricas quanto as relações entre as variáveis numéricas entre si.

```
# Criando um data frame com as variáveis  a serem analisadas
data1 <- data[,3:9]
plot(data1)
```



Analisando o gráfico de dispersão acima fica dificil de definir quais variaveis estão relacionada linearmente para conseguirmos definir um modelo de regressão linear, para isso vamos verificar a matriz de correlação

```
cor(data1[, -1], method = "pearson")
```

```
##                    price        pages       charpp   citations foundingyear
## price          1.00000000  0.493724318  0.074579257  0.02804096   0.25341618
## pages          0.49372432  1.000000000 -0.008986512  0.53700823  -0.15734335
## charpp         0.07457926 -0.008986512  1.000000000  0.10445760   0.03152999
## citations      0.02804096  0.537008232  0.104457600  1.00000000  -0.38303682
## foundingyear   0.25341618 -0.157343349  0.031529988 -0.38303682   1.00000000
## subs          -0.31196769  0.371405508  0.083192651  0.58469923  -0.40737210
##                      subs
## price         -0.31196769
## pages          0.37140551
## charpp         0.08319265
## citations      0.58469923
## foundingyear -0.40737210
## subs           1.00000000
```

Analisando a matriz de correlação de pearson, podemos ver que a variável subs é a que mais apresenta uma relação linear com as demais, mas logo abaixo vamos verificar a correlação de spearman adicionando a variável categórica *society* para verificar se a uma melhora na relação da correlação das variáveis.

```
data1$society_num <- ifelse(data1$society == "yes", 1, 0)
cor(data1[, -1], method = "spearman")
```

```
##                   price       pages      charpp   citations foundingyear
## price         1.00000000  0.32988587  0.04860780  0.06900815   0.36477761
## pages         0.32988587  1.00000000 -0.03338507  0.62468902  -0.17928897
## charpp        0.04860780 -0.03338507  1.00000000  0.14817360   0.02881994
## citations     0.06900815  0.62468902  0.14817360  1.00000000  -0.38936841
## foundingyear  0.36477761 -0.17928897  0.02881994 -0.38936841   1.00000000
## subs         -0.38282018  0.35807984 -0.01381436  0.62541702  -0.61734677
## society_num  -0.25323166  0.19179827  0.14070817  0.26656418  -0.20200107
##                    subs society_num
## price        -0.38282018  -0.2532317
## pages         0.35807984   0.1917983
## charpp       -0.01381436   0.1407082
## citations     0.62541702   0.2665642
## foundingyear -0.61734677  -0.2020011
## subs          1.00000000   0.2481580
## society_num   0.24815798   1.0000000
```

Podemos verificar que a correlação da variável society não é tão impactante, portanto vamos focar na analise das variáveis numéricas.

```
# Criando um data frame com as variaveis a serem analisadas
data1 <- data[,4:9]
```

Transformação

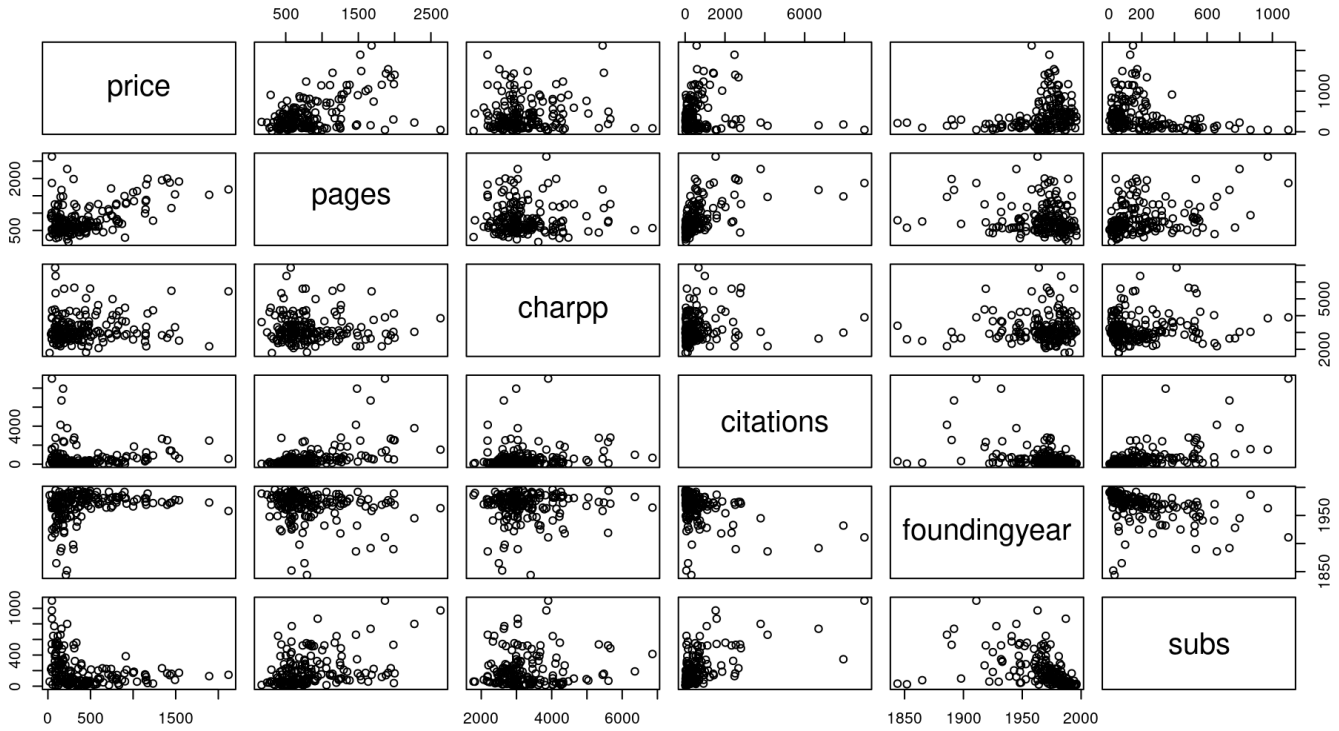
Vamos explorar a possibilidade de melhorar a relação entre as variáveis aplicando uma transformação logarítmica.

```
data1_log <- log(data1)
cor(data1_log)
```

```
##                   price      pages      charpp   citations foundingyear
## price         1.00000000  0.3346682  0.06207293  0.06437166   0.29777059
## pages         0.33466816  1.0000000 -0.01625150  0.64640656  -0.16382456
## charpp        0.06207293 -0.0162515  1.00000000  0.18835203   0.03159593
## citations     0.06437166  0.6464066  0.18835203  1.00000000  -0.32993488
## foundingyear  0.29777059 -0.1638246  0.03159593 -0.32993488   1.00000000
## subs         -0.34144809  0.3770614  0.02281673  0.64439947  -0.38463395
##                    subs
## price        -0.34144809
## pages         0.37706137
## charpp        0.02281673
## citations     0.64439947
## foundingyear -0.38463395
## subs          1.00000000
```

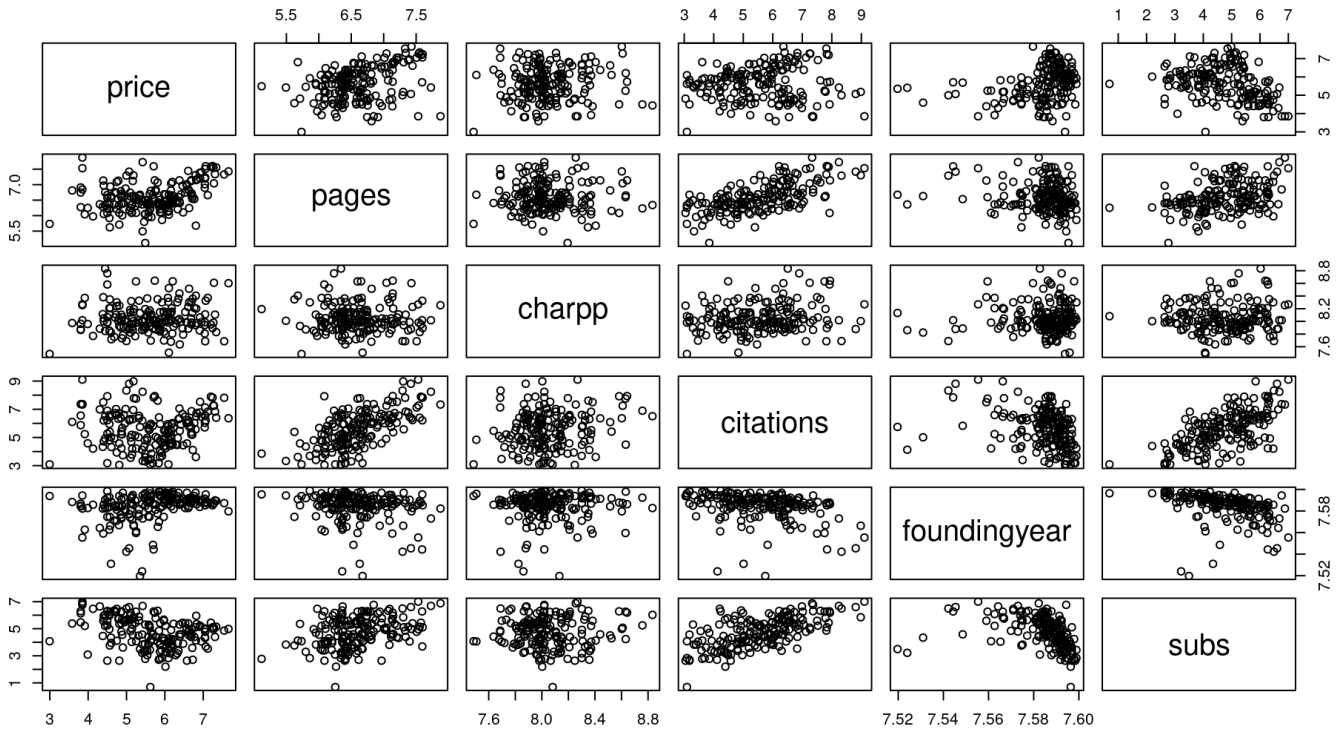
```
plot(data1, main = "Gráfico de dispersão dados normais")
```

Gráfico de dispersão dados normais




```
plot(data1_log, main = "Gráfico de dispersão dados transformado (log)")
```

Gráfico de dispersão dados transformado (log)



Modelos regressivos

Apesar da transformação logarítmica, não observamos uma melhora significativa na relação linear entre as variáveis, o que sugere que outras abordagens podem ser necessárias para capturar melhor a relação entre elas. Além disso, é importante notar que as variáveis citations e pages estão altamente correlacionadas.

Para investigar mais a fundo, vamos propor dois modelos:

Modelo 1 (fit1): Utilizando os dados sem transformação.

Modelo 2 (fit2): Utilizando os dados após a transformação logarítmica, bem como a inclusão da interação entre pages e citations.

Além disso, para a construção do nosso modelo, optaremos por selecionar a variável subs como nossa variável resposta. Essa escolha é fundamentada na observação de uma relação mais forte entre subs e as outras variáveis, sugerindo que subs pode ser a variável dependente que estamos interessados em prever.

```
pc = data1$citations / data1$pages
fit1 <- lm(data1$subs ~ data1$price + pc + data1$charpp)
fit2 <- lm(data1_log$subs ~ data1_log$price + log(pc) + data1_log$charpp)
```

```
summary(fit1)
```

```
##
## Call:
## lm(formula = data1$subs ~ data1$price + pc + data1$charpp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -487.59 -108.04  -28.86   63.32  738.44
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  201.652474  49.757238   4.053 7.59e-05 ***
## data1$price   -0.138614   0.031348  -4.422 1.71e-05 ***
## pc           127.705640  13.989941   9.128  < 2e-16 ***
## data1$charpp  -0.009307   0.015196  -0.612    0.541
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 160.4 on 176 degrees of freedom
## Multiple R-squared:  0.3951, Adjusted R-squared:  0.3848
## F-statistic: 38.32 on 3 and 176 DF,  p-value: < 2.2e-16
```

```
summary(fit2)
```

```
##
## Call:
## lm(formula = data1_log$subs ~ data1_log$price + log(pc) + data1_log$charpp)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.63206 -0.51268 -0.05475  0.53187  2.19709
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      11.69788    2.21429   5.283 3.72e-07 ***
## data1_log$price  -0.35034    0.06514  -5.378 2.37e-07 ***
## log(pc)           0.65414    0.05777  11.323  < 2e-16 ***
## data1_log$charpp -0.53638    0.27298  -1.965    0.051 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8092 on 176 degrees of freedom
## Multiple R-squared:   0.49,  Adjusted R-squared:  0.4813
## F-statistic: 56.37 on 3 and 176 DF,  p-value: < 2.2e-16
```

Afim de melhorar o R^2 ajustado do modelo, vamos tirar o intercepto

```
fit1 <- lm(data1$subs ~ -1 + data1$price + pc + data1$charpp)
fit2 <- lm(data1_log$subs ~ -1 + data1_log$price + log(pc) + data1_log$charpp)
```

```
summary(fit1)
```

```
##
## Call:
## lm(formula = data1$subs ~ -1 + data1$price + pc + data1$charpp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -491.67  -93.24  -25.01   81.96  723.98
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## data1$price   -0.115167   0.032124  -3.585 0.000436 ***
## pc           125.168429  14.572171   8.590 4.42e-15 ***
## data1$charpp   0.046951   0.006447   7.283 1.03e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 167.3 on 177 degrees of freedom
## Multiple R-squared:  0.6576, Adjusted R-squared:  0.6518
## F-statistic: 113.3 on 3 and 177 DF,  p-value: < 2.2e-16
```

```
summary(fit2)
```

```
##
## Call:
## lm(formula = data1_log$subs ~ -1 + data1_log$price + log(pc) +
##     data1_log$charpp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8293 -0.5411  0.0329  0.6068  2.1707
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## data1_log$price  -0.32179    0.06968  -4.618 7.43e-06 ***
## log(pc)           0.57659    0.05997   9.614  < 2e-16 ***
## data1_log$charpp  0.88532    0.04912  18.024  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8685 on 177 degrees of freedom
## Multiple R-squared:  0.9687, Adjusted R-squared:  0.9682
## F-statistic:  1828 on 3 and 177 DF,  p-value: < 2.2e-16
```

É interessante observar que o R^2 ajustado do modelo sem intercepto apresentou uma melhora considerável. Além disso, o modelo fit2, que inclui a transformação logarítmica, alcançou um R^2 de 0.9765. Esses resultados sugerem que as variáveis podem de fato ter uma relação linear, o que fortalece a validade do modelo proposto.

```
y <- data1_log$subs
x <- as.matrix(fit2$model[,2:ncol(fit2$model)])
B <- t(as.matrix(solve(t(x) %*% x) %*% t(x) %*% y))
H <- x %*% solve((t(x) %*% x)) %*% t(x)
```

```
cor(x)
```

```
##                  data1_log$price     log(pc) data1_log$charpp
## data1_log$price       1.00000000 -0.06909207       0.06207293
## log(pc)              -0.06909207  1.00000000       0.23836031
## data1_log$charpp      0.06207293  0.23836031       1.00000000
```

Após a divisão entre a variável *citations* e *pages*, que fornece uma medida de quantas citações há por página, observamos uma melhoria na correlação entre as variáveis. Isso sugere que não há evidências significativas de multicolinearidade entre elas. Além disso, a presença da matriz HAT (H) reforça a possível inexistência de multicolinearidade, como podemos verificar abaixo.

```
# Print resumido da matrix Hat
print(as.matrix(H[1:10,1:7]))
```

```
##             1          2          3          4          5          6          7
## 1  0.03334818 0.03919208 0.02475279 0.02783288 0.03041563 0.02653104 0.03664958
## 2  0.03919208 0.05563823 0.01860552 0.02613073 0.02757749 0.02251747 0.04386381
## 3  0.02475279 0.01860552 0.02988460 0.02788107 0.03160276 0.02921026 0.02638650
## 4  0.02783288 0.02613073 0.02788107 0.02776065 0.03104506 0.02811376 0.03007296
## 5  0.03041563 0.02757749 0.03160276 0.03104506 0.03491885 0.03167158 0.03287591
## 6  0.02653104 0.02251747 0.02921026 0.02811376 0.03167158 0.02897736 0.02849188
## 7  0.03664958 0.04386381 0.02638650 0.03007296 0.03287591 0.02849188 0.04041623
## 8  0.02837265 0.02665774 0.02842024 0.02829691 0.03168122 0.02866149 0.03068797
## 9  0.01878232 0.01894635 0.01720283 0.01773172 0.01939483 0.01760797 0.02014702
## 10 0.02084542 0.01850926 0.02194980 0.02146689 0.02392352 0.02192132 0.02229891
```

```
df <- fit2$df.residual
n <- nrow(data1_log)
p <- ncol(fit2$model) - 1

ssreg <- B %*% t(x) %*% y
sstot <- t(y) %*% y
ssres <- sstot - ssreg

msres <- as.matrix(ssres / (n - p))
var_B1 <- msres * solve(t(x) %*% x)[1,1]
var_B2 <- msres * solve(t(x) %*% x)[2,2]
var_B3 <- msres * solve(t(x) %*% x)[3,3]

r <- ssreg / sstot
r_j <-  1- ((ssres / (n - p)) / (sstot / (n - 1)))


# Criar uma matriz para armazenar os intervalos de confiança
ic_B <- matrix(NA, nrow = 3, ncol = 2)

# Nomear as linhas e colunas da matriz
rownames(ic_B) <- c("B1", "B2", "B3")
colnames(ic_B) <- c("Limite Inferior", "Limite Superior")

# Intervalos de confianças para os parâmetros B1, B2, B3, B4:
ic_B[1,1] <- B[1] + qt(0.025, df) * sqrt(var_B1)
ic_B[1,2] <- B[1] + qt(0.975, df) * sqrt(var_B1)

ic_B[2,1] <- B[2] + qt(0.025, df) * sqrt(var_B2)
ic_B[2,2] <- B[2] + qt(0.975, df) * sqrt(var_B2)


ic_B[3,1] <- B[3] + qt(0.025, df) * sqrt(var_B2)
ic_B[3,2] <- B[3] + qt(0.975, df) * sqrt(var_B2)


# Exibir a matriz
print(ic_B)
```

```
##    Limite Inferior Limite Superior
## B1      -0.4592962      -0.1842766
## B2       0.4582398       0.6949438
## B3       0.7669663       1.0036703
```

```
# Estimativa pontual para os parâmetros
rownames(B) <- c("Estimativa Pontual")
print(t(B))
```

```
##                  Estimativa Pontual
## data1_log$price          -0.3217864
## log(pc)                   0.5765918
## data1_log$charpp          0.8853183
```

```
# Intervalo de confiança para sigma² com um nivel de significância de 5%
ic_SIGMA2 <- matrix(NA, nrow = 1, ncol = 2)
rownames(ic_SIGMA2) <- c("σ²")
colnames(ic_SIGMA2) <- c("Limite Inferior", "Limite Superior")
ic_SIGMA2[1, 1] <- df*msres/qchisq(0.975, df)
ic_SIGMA2[1, 2] <- df*msres/qchisq(0.025, df)

# Estimativa intervala para o sigma² com um nivel de significância de 5%
ic_SIGMA2
```

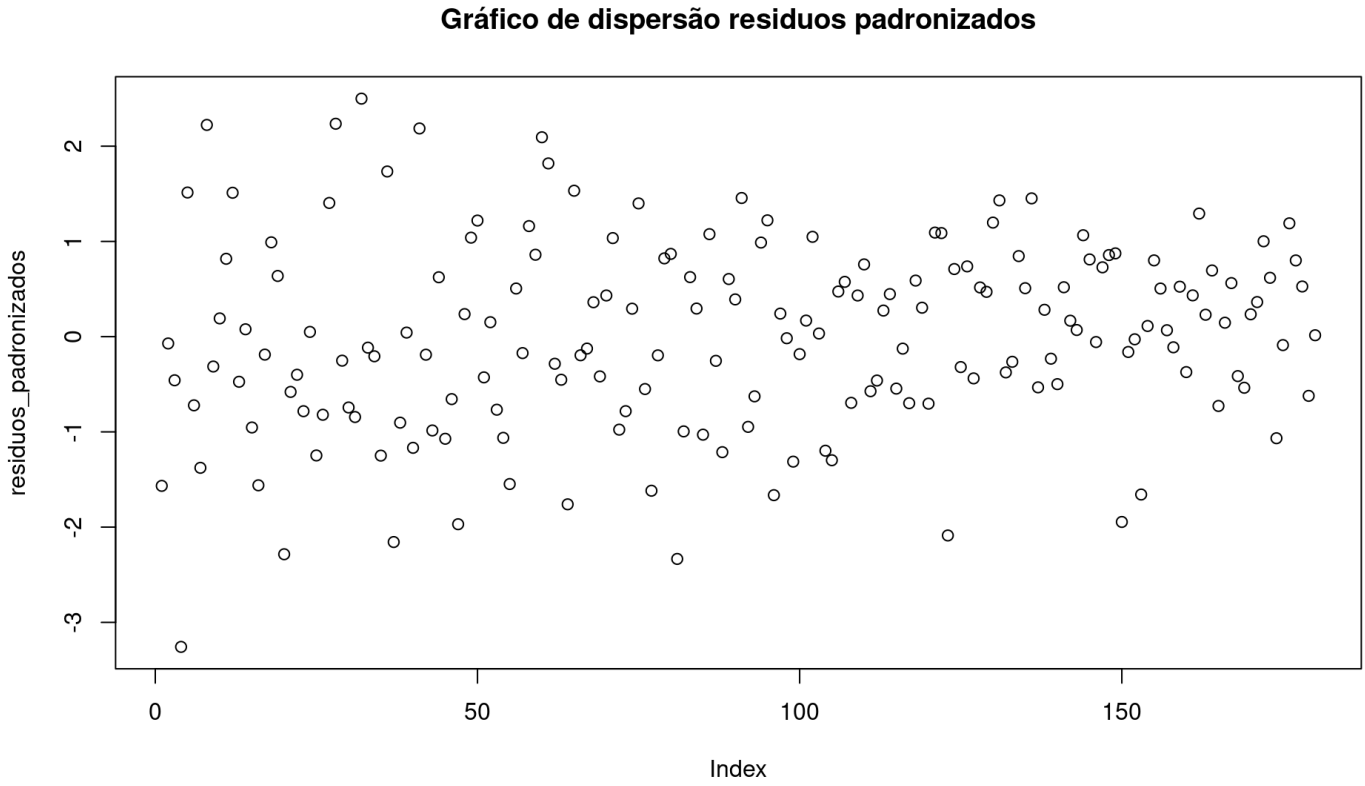
```
##    Limite Inferior Limite Superior
## σ²        0.618931       0.9399588
```

```
# Estimativa pontual para sigma²
rownames(msres) <- c("σ²")
colnames(msres) <- c("Estimativa Pontual")
msres
```

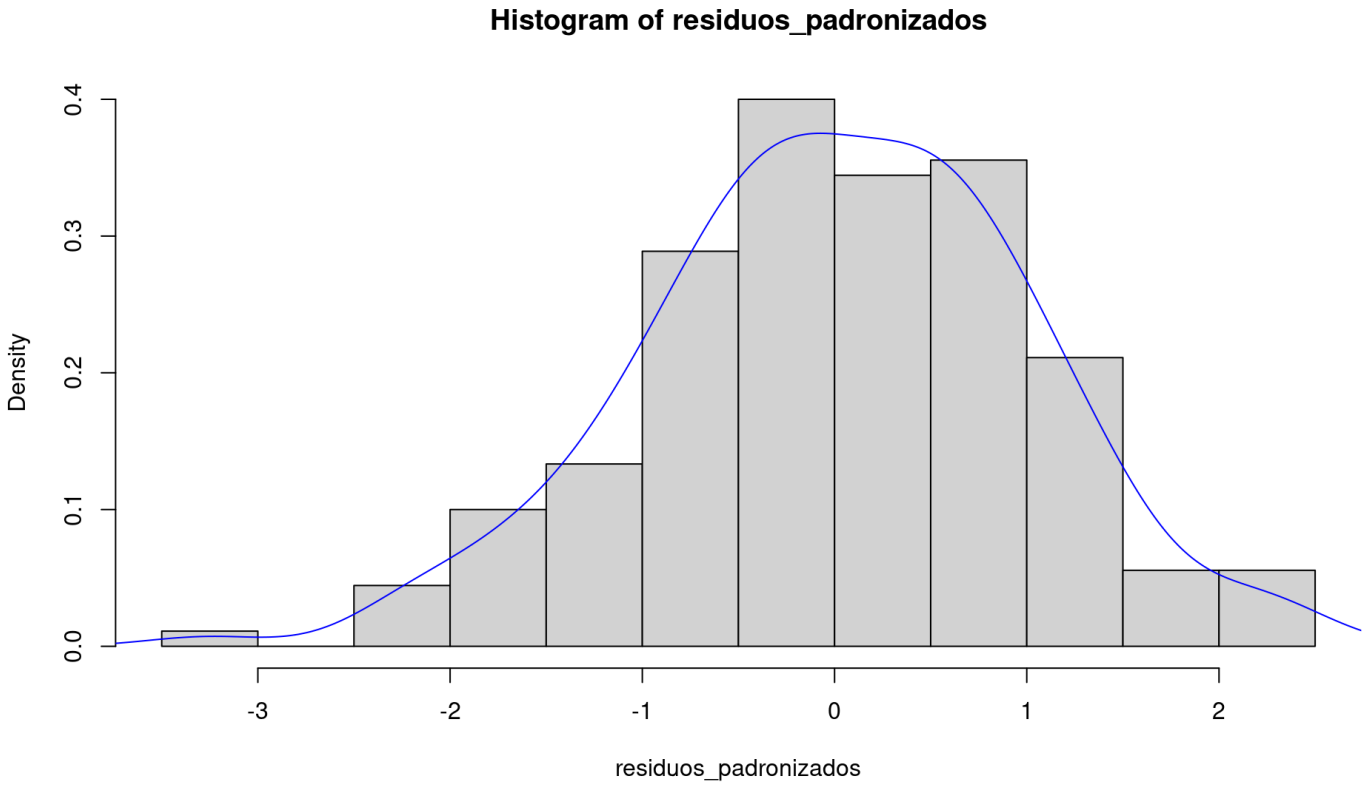
```
##    Estimativa Pontual
## σ²          0.7543713
```

Analise de resíduo

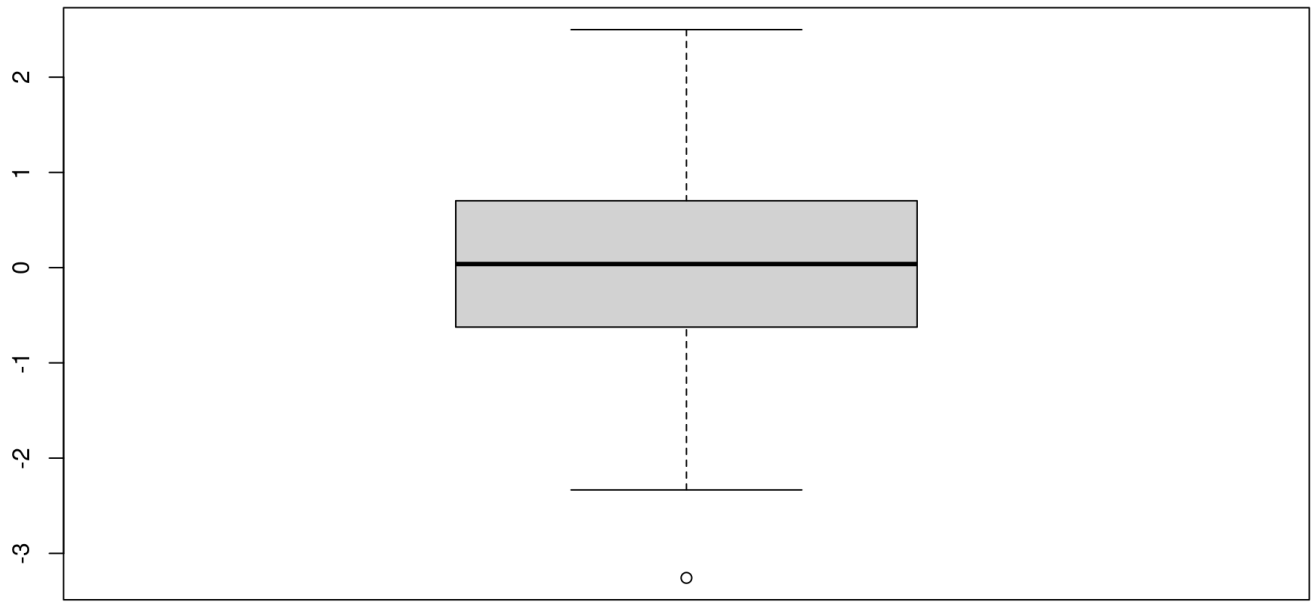
```
# Residuos padronizados

residuos_padronizados <- (data1_log$subs - fit2$fitted.values) / as.vector(sqrt(msres))
plot(residuos_padronizados, main = "Gráfico de dispersão residuos padronizados")
```



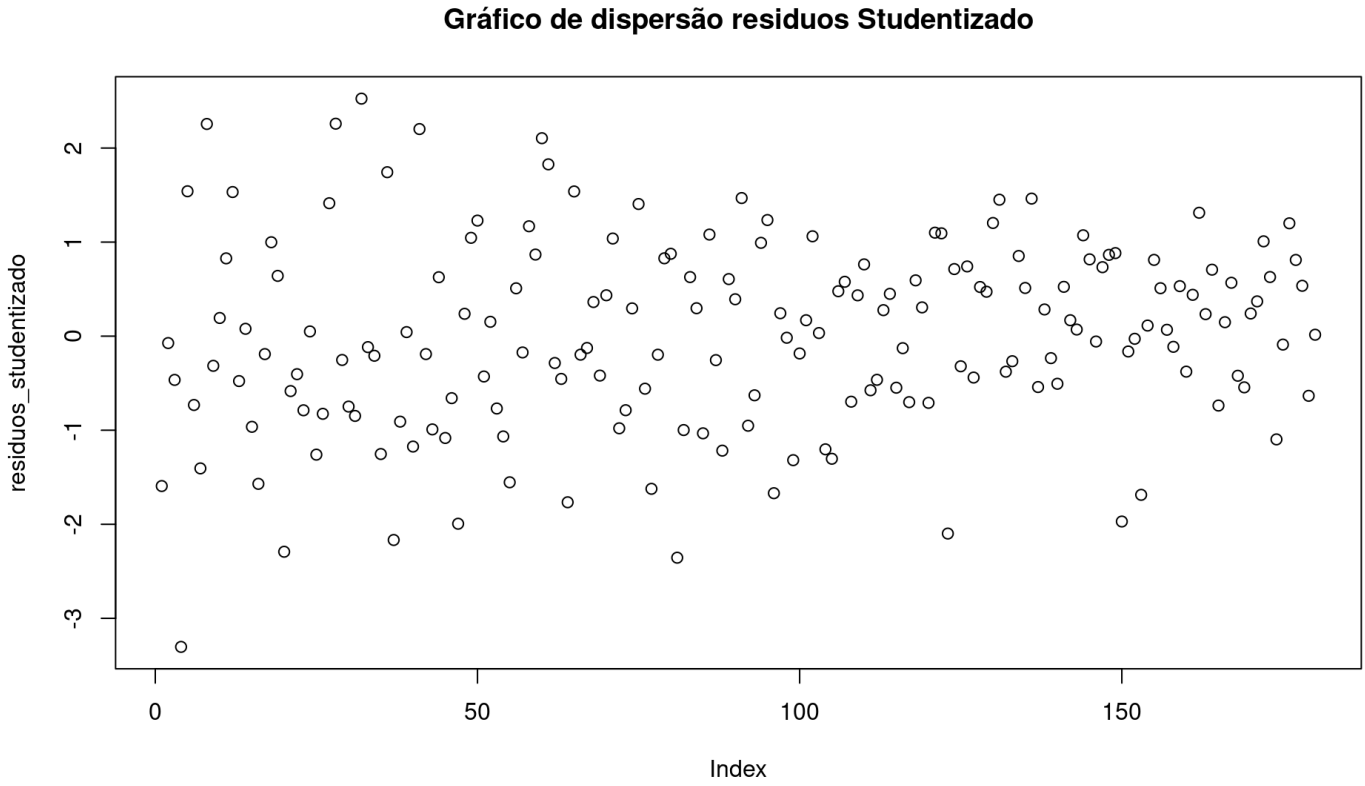

```
hist(residuos_padronizados, freq=FALSE)
lines(density(residuos_padronizados), col='blue')
```



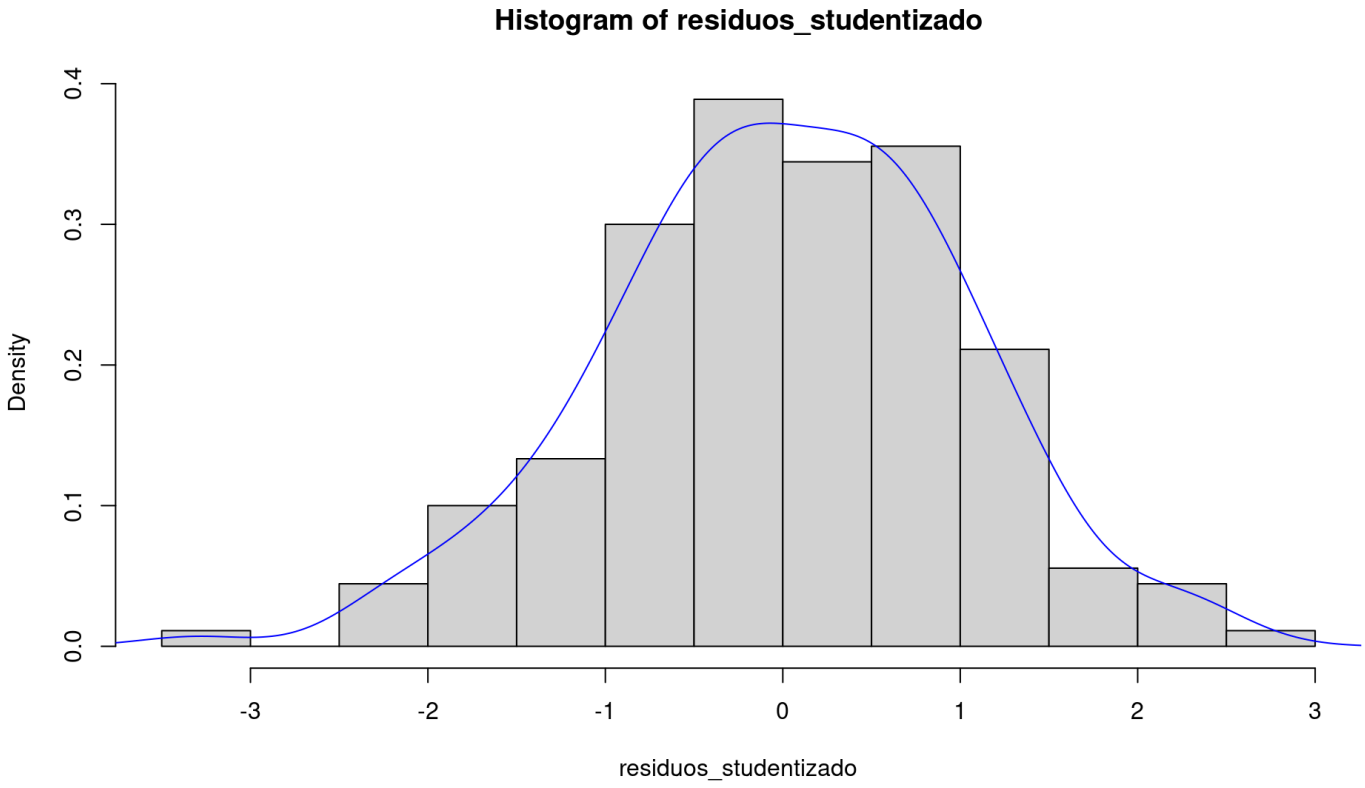

```
boxplot(residuos_padronizados, main = 'Box-Plot: Residuos Padronizados')
```



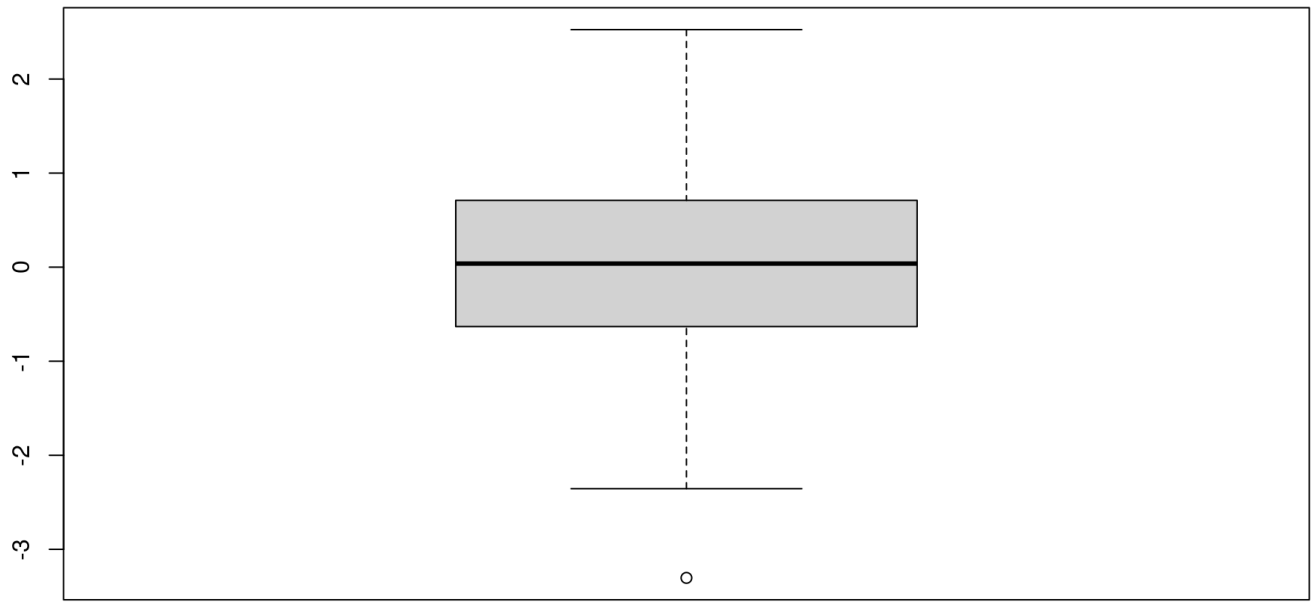

```
# Residuos studentizados

residuos_studentizado <- (data1_log$subs - fit2$fitted.values) / as.vector(sqrt(msres[1]
* (1 - diag(H))))
plot(residuos_studentizado, main = "Gráfico de dispersão residuos Studentizado")
```




```
hist(residuos_studentizado, freq=FALSE)
lines(density(residuos_studentizado), col='blue')
```




```
boxplot(residuos_studentizado, main = 'Box-Plot: Resíduos Studentizado')
```




```
# inveralo dos x's
print('x1:')
```

```
## [1] "x1:"
```

```
c(min(x[data1_log$price]),max(x[data1_log$price]))
```

```
## [1] 2.995732 6.093570
```

```
print('x2:')
```

```
## [1] "x2:"
```

```
c(min(x[data1_log$citations]),max(x[data1_log$citations]))
```

```
## [1] 4.49981 6.09357
```

```
print('x3:')
```

```
## [1] "x3:"
```

```
c(min(x[data1_log$foundingyear]),max(x[data1_log$foundingyear]))
```

```
## [1] 4.49981 4.49981
```

```
print('x4:')
```

```
## [1] "x4:"
```

```
c(min(x[data1_log$pages]),max(x[data1_log$pages]))
```

```
## [1] 4.499810 5.840642
```

```
x0 <- t(t(c(6, 6, 4.4)))
extrapolacao <- t(x0) %*% solve((t(x) %*% x)) %*% x0
if (extrapolacao > max(H)) {
  print("As novas observações são uma extrapolação")
} else {
  print("Não é uma extrapolação")
}
```

```
## [1] "As novas observações são uma extrapolação"
```

```
# Propondo modelo reduzido 1 (sem pages)
fit_red <- lm(data1_log$subs ~ -1 + data1_log$price + data1_log$citations + data1_log$fo
undingyear)
summary(fit_red)
```

```
##
## Call:
## lm(formula = data1_log$subs ~ -1 + data1_log$price + data1_log$citations +
##     data1_log$foundingyear)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.64240 -0.50825 -0.00988  0.48809  1.95397
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## data1_log$price        -0.46456    0.06019  -7.718 8.31e-13 ***
## data1_log$citations     0.56838    0.04208  13.506  < 2e-16 ***
## data1_log$foundingyear  0.55153    0.05321  10.365  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.748 on 177 degrees of freedom
## Multiple R-squared:  0.9768, Adjusted R-squared:  0.9764
## F-statistic:  2486 on 3 and 177 DF,  p-value: < 2.2e-16
```

```
x_red <- as.matrix(fit_red$model[,2:ncol(fit_red$model)])
B_red <- t(as.matrix(solve(t(x_red) %*% x_red) %*% t(x_red) %*% y))
ssreg_red <- B_red %*% t(x_red) %*% y
ssres_res = sstot - ssreg_red
anova
```

```
## function (object, ...)
## UseMethod("anova")
## <bytecode: 0x58e4c1d03d58>
## <environment: namespace:stats>
```

```
ssextra = ssreg - ssreg_red

# Teste de hipotese para verificar se o ganho é relevante
f0 <- ssextra/(ssres / (p - 1))
p_f = pf(f0, 1, (p-1))

p_f
```

```
##      [,1]
## [1,]    0
```

```
# Propondo modelo reduzido 2 (citations = foundingyear

# Propondo modelo reduzido 1 (sem pages)
w = (data1_log$citations + data1_log$foundingyear)
fit_red1 <- lm(data1_log$subs ~ -1 + data1_log$price + w)
summary(fit_red1)
```

```
##
## Call:
## lm(formula = data1_log$subs ~ -1 + data1_log$price + w)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.66140 -0.51728 -0.00259  0.49016  1.93898
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## data1_log$price -0.47037    0.05272  -8.923 5.46e-16 ***
## w                0.56124    0.02274  24.677  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.746 on 178 degrees of freedom
## Multiple R-squared:  0.9768, Adjusted R-squared:  0.9765
## F-statistic:  3749 on 2 and 178 DF,  p-value: < 2.2e-16
```

```
x_red1 <- as.matrix(fit_red1$model[,2:ncol(fit_red1$model)])
B_red1 <- t(as.matrix(solve(t(x_red1) %*% x_red1) %*% t(x_red1) %*% y))
ssreg_red1 <- B_red1 %*% t(x_red1) %*% y

ssextra = ssreg - ssreg_red1

# Teste de hipotese para verificar se o ganho é relevante
f0 <- ssextra/(ssres / (p - 1))
p_f = pf(f0, 1, (p-1))

p_f
```

```
##      [,1]
## [1,]    0
```

```
# Verificação multicolinearidade


# Página e Citação possui uma correlação linear elevada, 0.6464066
c = solve(t(x) %*% x)


# Dado os fatores de inflação de variância, não tem multicolinearidade
vif1 <- c[1,1]
vif2 <- c[2,2]
vif3 <- c[3,3]

vif1
```

```
## [1] 0.006436159
```

```
vif2
```

```
## [1] 0.00476772
```

```
vif3
```

```
## [1] 0.003198384
```