

MAE0217 - Estatística Descritiva - Lista 4

Natalia Hitomi Koza¹
Rafael Gonçalves Pereira da Silva²
Ricardo Geraldies Tolesano³
Rubens Kushimizo Rodrigues Xavier⁴
Rubens Gomes Neto⁵
Rubens Santos Andrade Filho⁶
Thamires dos Santos Matos⁷

June de 2021

Sumário

Exercício 1	2
Exercício 2	24
Exercício 3	24
Exercício 4	24
Exercício 15	24
Exercício 16	24

¹Número USP: 10698432

²Número USP: 9009600

³Número USP: 10734557

⁴Número USP: 8626718

⁵Número USP: 9318484

⁶Número USP: 10370336

⁷Número USP: 9402940

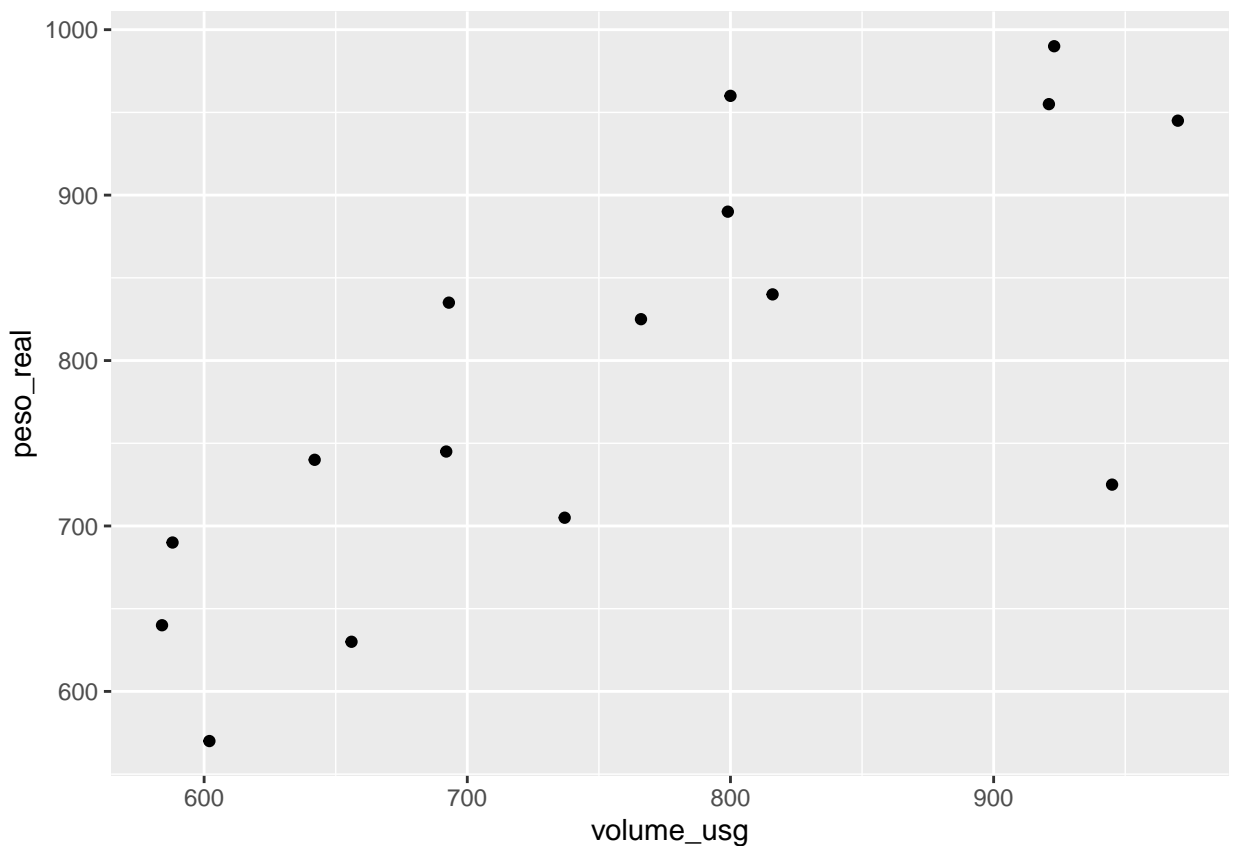
Exercício 1

i)

Tomaremos Volume USG como a variável explicativa x e Peso Real como a variável resposta y . Adotaremos o modelo de regressão linear simples $y_i = \alpha + \beta x_i + e_i$, onde α é o intercepto, β é a inclinação da reta, e e_i são erros aleatórios não correlacionados.

ii)

```
dados1 <- read_excel("data/peso_volume_figado.xlsx")
dados1 <- dados1[order(dados1$volume_usg), ]
# ggplot(dados, aes(x=volume_usg, y=peso_real)) + geom_point() + geom_smooth(method=lm)
ggplot(dados1, aes(x=volume_usg, y=peso_real)) + geom_point()
```



iii)

Realizaremos o ajuste do modelo e mostraremos algumas métricas de qualidade do modelo:

```
ajustarModelo <- function(dados) {
  ajuste <- lm(peso_real ~ volume_usg, data=dados)
  intercept <- ajuste$coefficients[1]
```

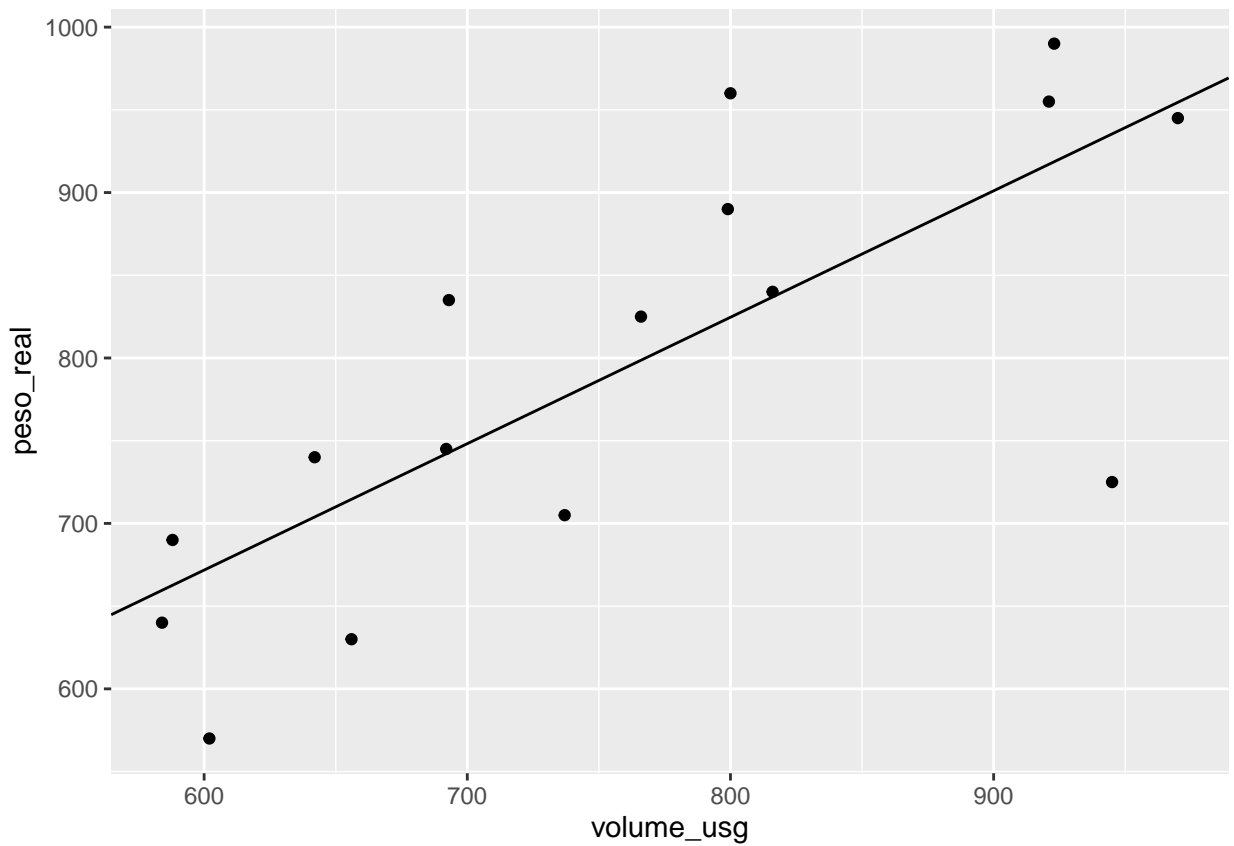
```

slope <- ajuste$coefficients[2]
p <- ggplot(dados, aes(x=volume_usg, y=peso_real)) + geom_point() + geom_abline(intercept = intercept, slope = slope)
plot(p)
print(summary(ajuste))
plot(ajuste)

return(ajuste)
}

ajuste <- ajustarModelo(dados1)

```

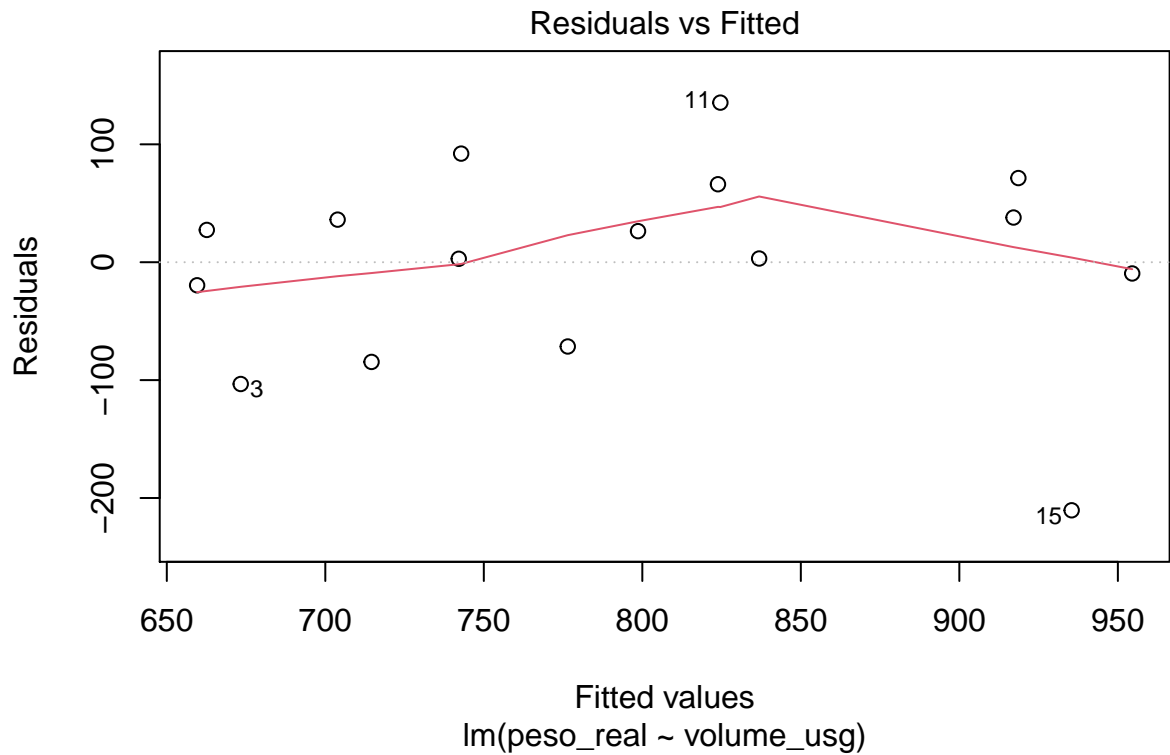


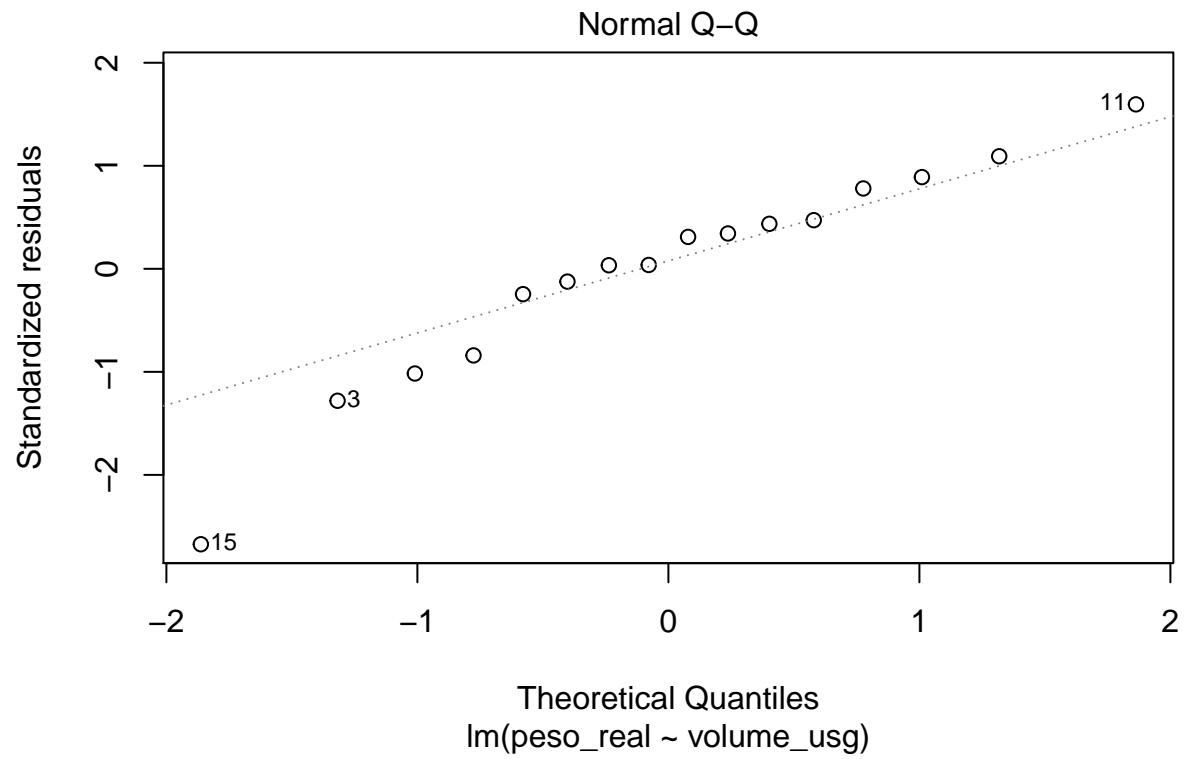
```

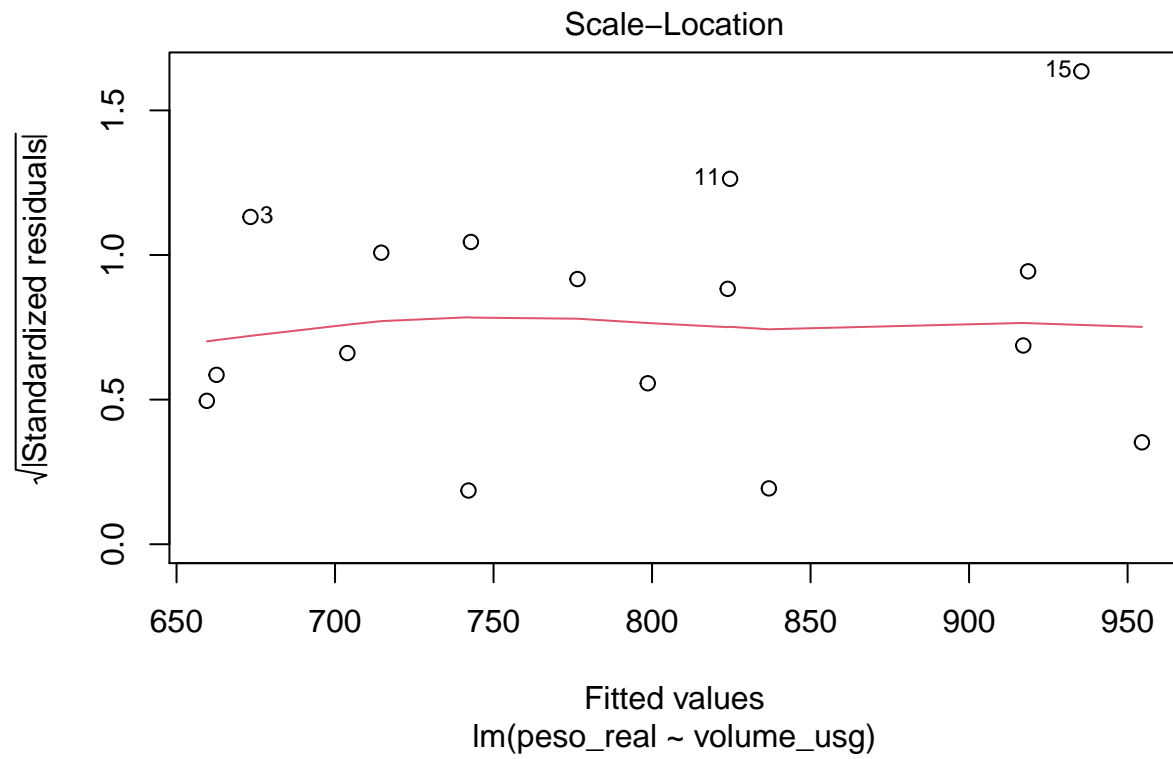
##
## Call:
## lm(formula = peso_real ~ volume_usg, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -210.43  -32.54   14.76   44.97  135.38
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  213.2762   133.3334   1.600  0.132011
## volume_usg    0.7642    0.1734   4.407  0.000597 ***
## ---

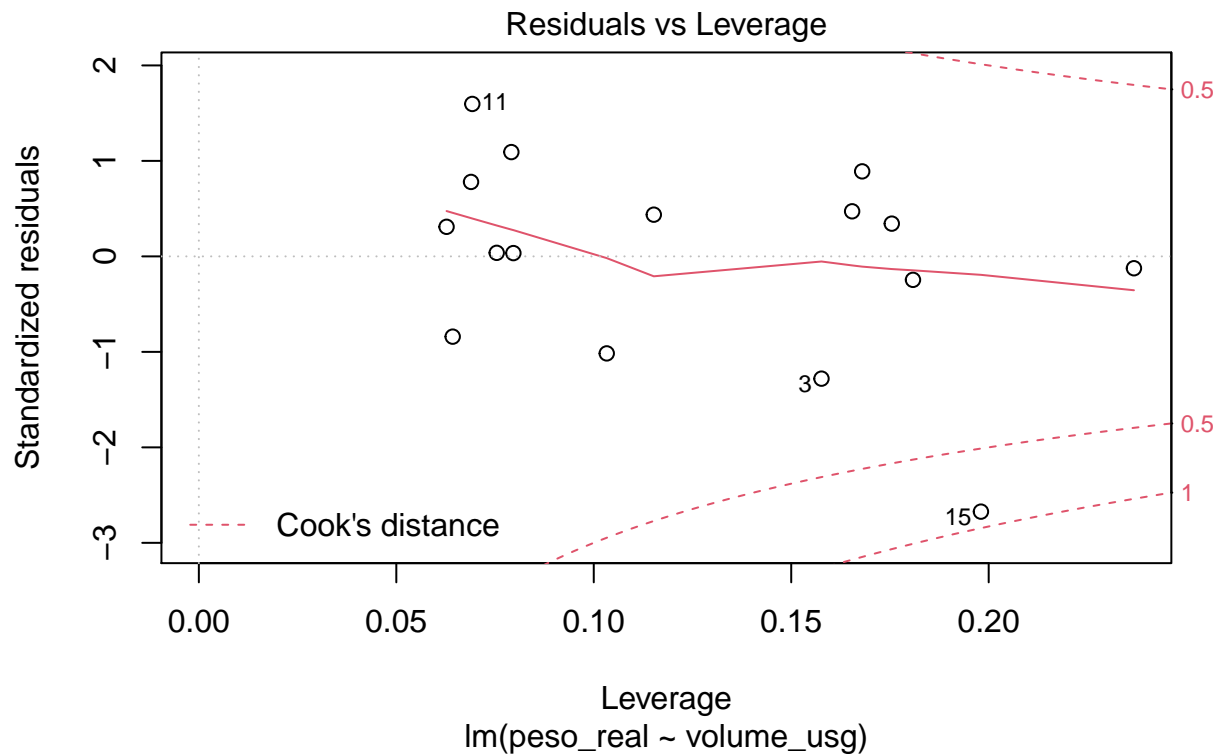
```

```
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 87.91 on 14 degrees of freedom
## Multiple R-squared:  0.5811, Adjusted R-squared:  0.5512
## F-statistic: 19.42 on 1 and 14 DF,  p-value: 0.000597
```



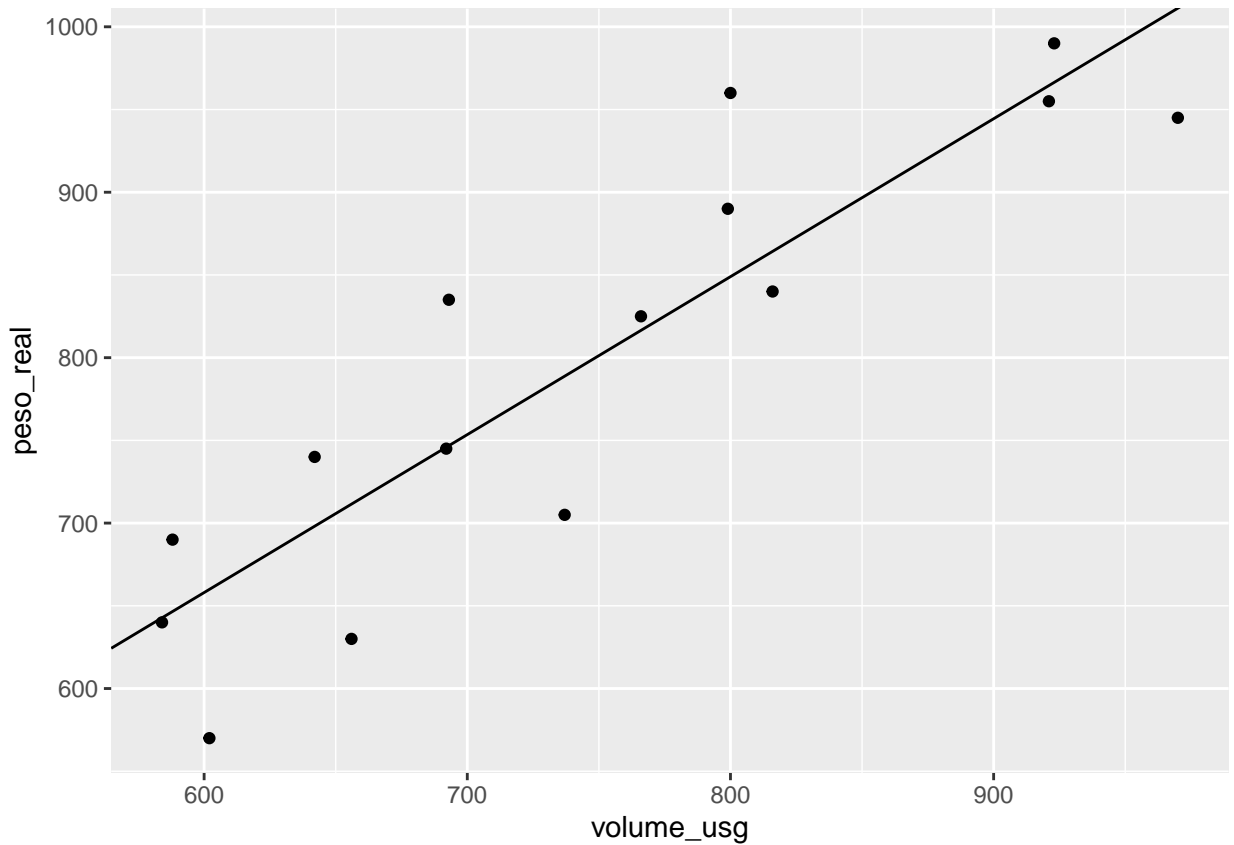




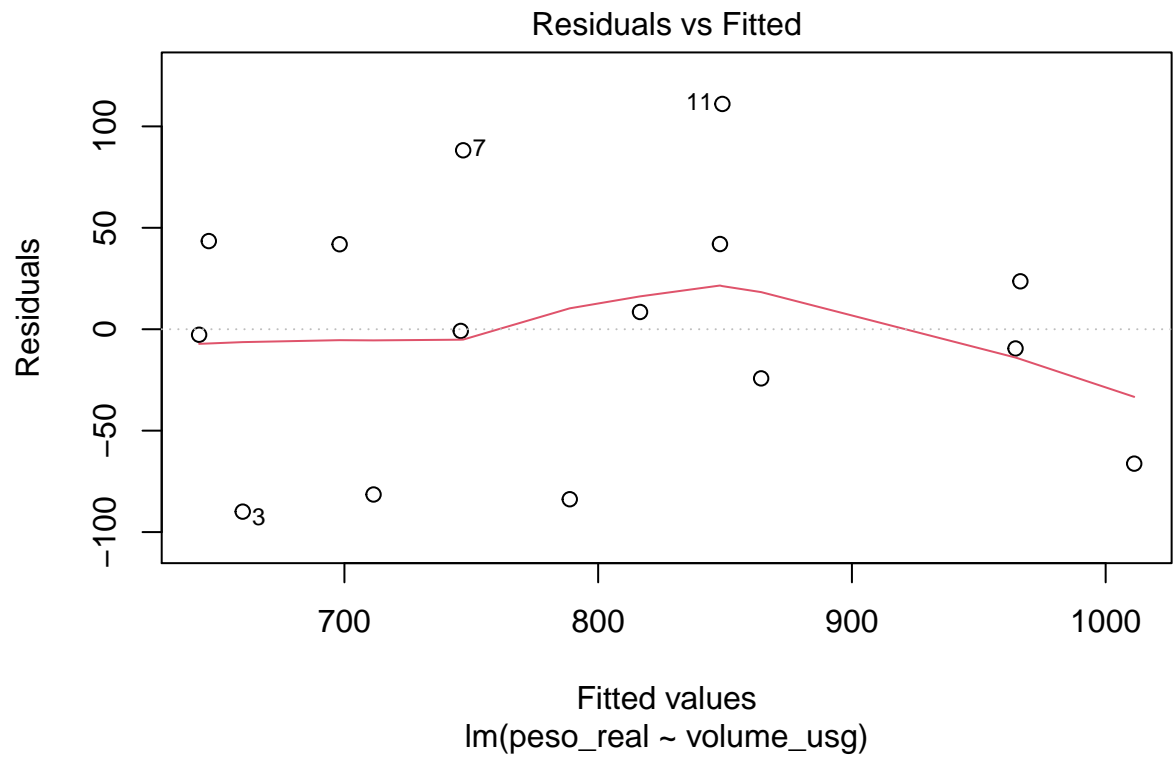


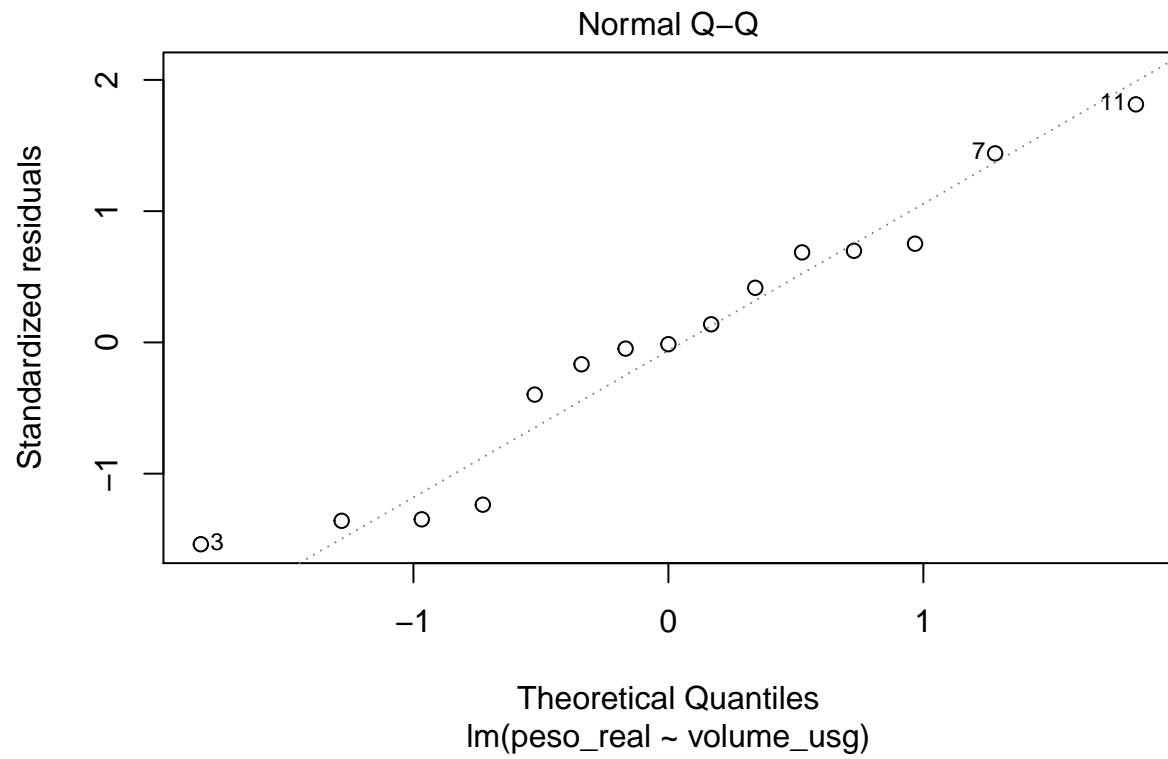
A análise do ajuste indicou que as observações 3, 11 e 15 são mais influentes no modelo. Em especial, a observação 15 se destaca como outlier em todos os gráficos mostrados. Realizaremos novamente o ajuste com essa observação removida. Não removeremos as observações 3 e 11 dado que possuímos poucas observações e elas não fogem do padrão na mesma intensidade elevada da observação 15.

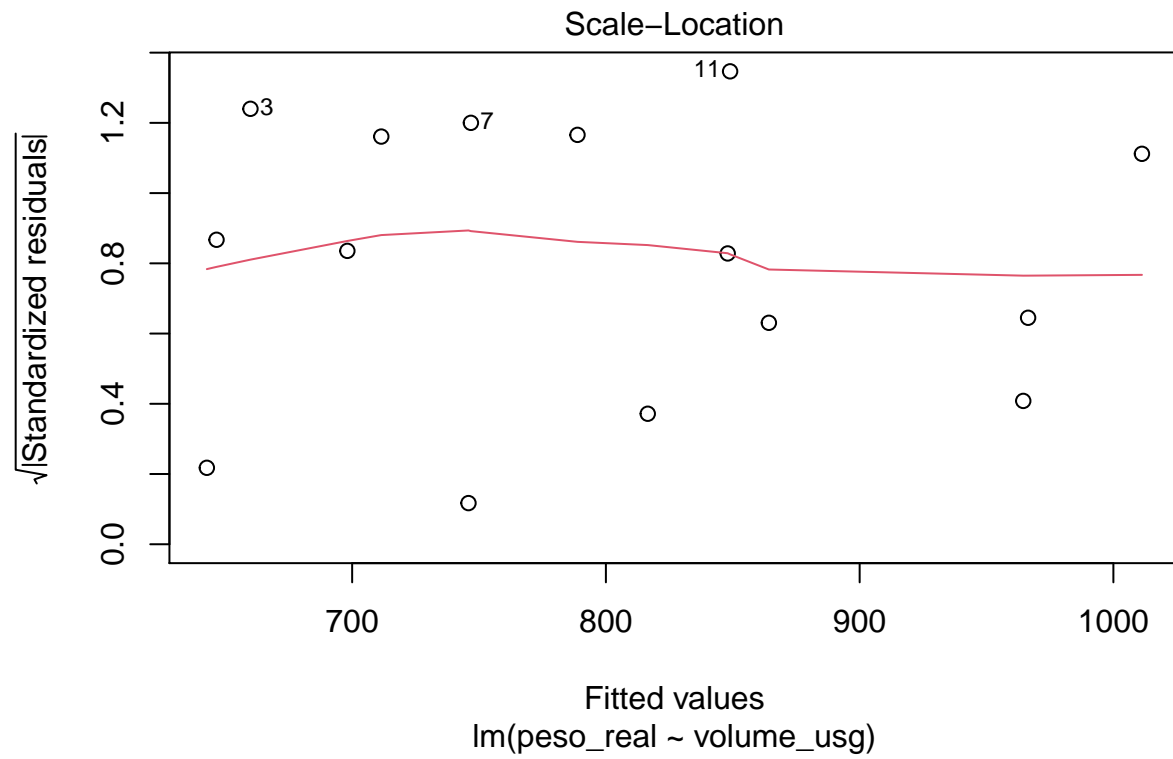
```
dados2 <- dados1[-c(15), ]
ajuste <- ajustarModelo(dados2)
```

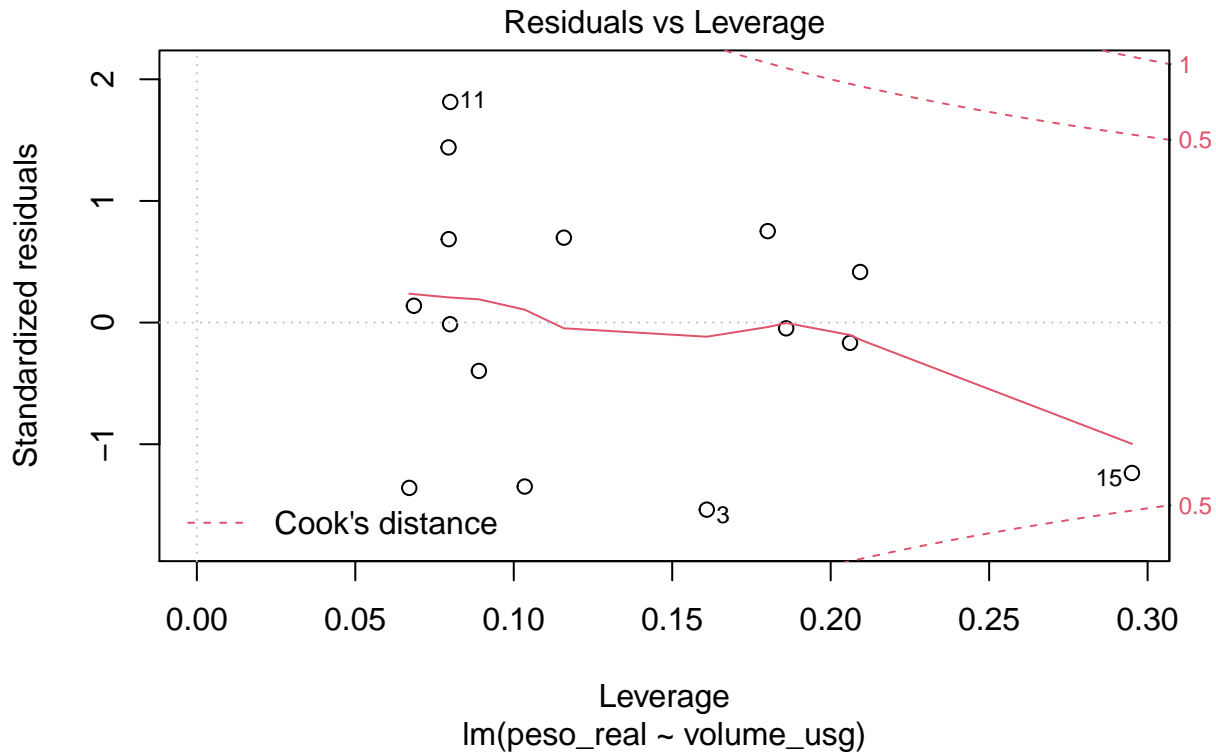


```
##
## Call:
## lm(formula = peso_real ~ volume_usg, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -89.914 -45.244  -0.841  41.949 111.047
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  85.1593   102.8848   0.828   0.423
## volume_usg    0.9547     0.1361   7.013 9.17e-06 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 63.83 on 13 degrees of freedom
## Multiple R-squared:  0.7909, Adjusted R-squared:  0.7748
## F-statistic: 49.18 on 1 and 13 DF, p-value: 9.167e-06
```







Observamos uma melhora significativa no valor R^2 após a remoção da observação 15. Os gráficos indicam que os resíduos possuem os valores dentro do esperado. Idealmente, o R^2 deveria estar próximo de 1, mas não está. Dessa forma, podemos concluir que o ajuste do modelo aproxima os dados, mas não estritamente. Assim, espera-se que o intervalo de confiança seja grande.

iv)

Construindo intervalos de confiança dos parâmetros:

```
confint(ajuste)
```

```
##              2.5 %      97.5 %
## (Intercept) -137.1098633 307.428386
## volume_usg   0.6606159  1.248869
```

v)

A seguir, construiremos a tabela.

```
volumes <- c(600, 700, 800, 900, 1000)
df <- data.frame(volume_usg = volumes)
previsto <- predict(ajuste, df, interval='confidence')
previsto <- data.frame(previsto)
```

```

intervalo <- previsto$fit - previsto$lwr
previsto <- cbind(volume_usg = volumes, previsto, intervalo = intervalo)
kable(previsto)

```

volume_usg	fit	lwr	upr	intervalo
600	658,00	602,24	713,77	55,77
700	753,48	715,40	791,56	38,08
800	848,95	809,96	887,95	39,00
900	944,43	886,80	1.002,06	57,63
1.000	1.039,90	957,12	1.122,68	82,78

vi)

vi)i)

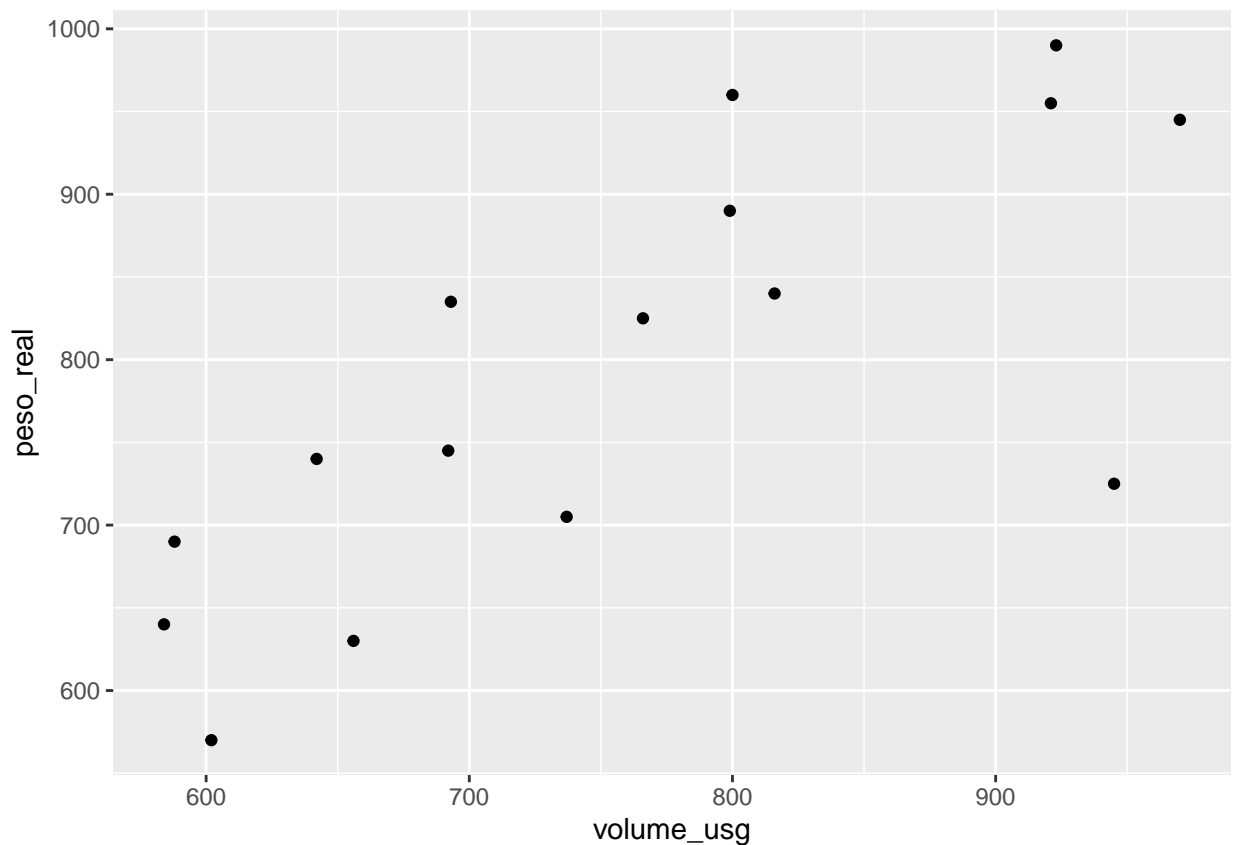
Novamente, tomaremos o Volume USG como a variável explicativa x e o Peso Real como a variável resposta y . Adotaremos o modelo de regressão linear simples $y_i = \beta x_i + e_i$, onde β é a inclinação da reta e e_i são erros aleatórios não correlacionados.

vi)ii)

```

dados3 <- data.frame(dados1)
ggplot(dados3, aes(x=volume_usg, y=peso_real)) + geom_point()

```

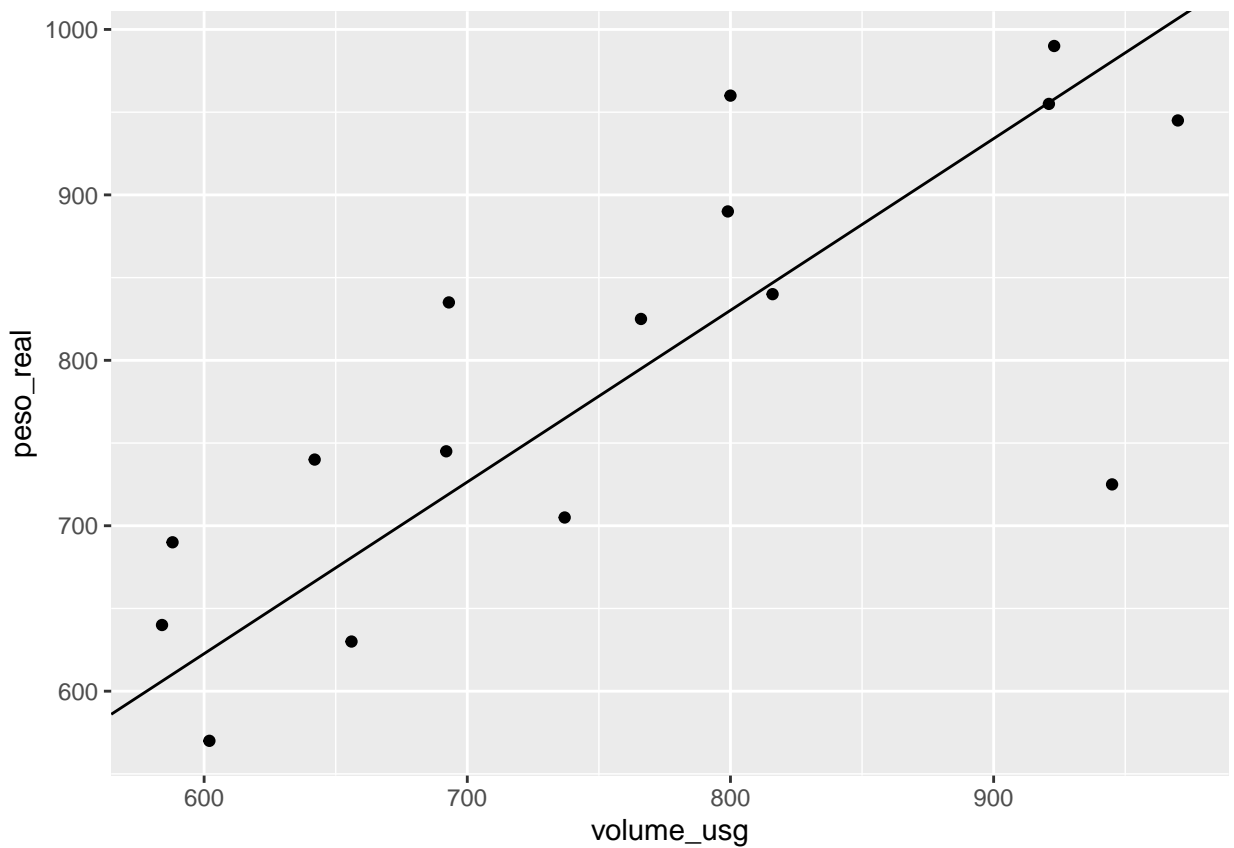


vi)iii)

Realizaremos o ajuste do modelo e mostraremos algumas métricas de qualidade do modelo:

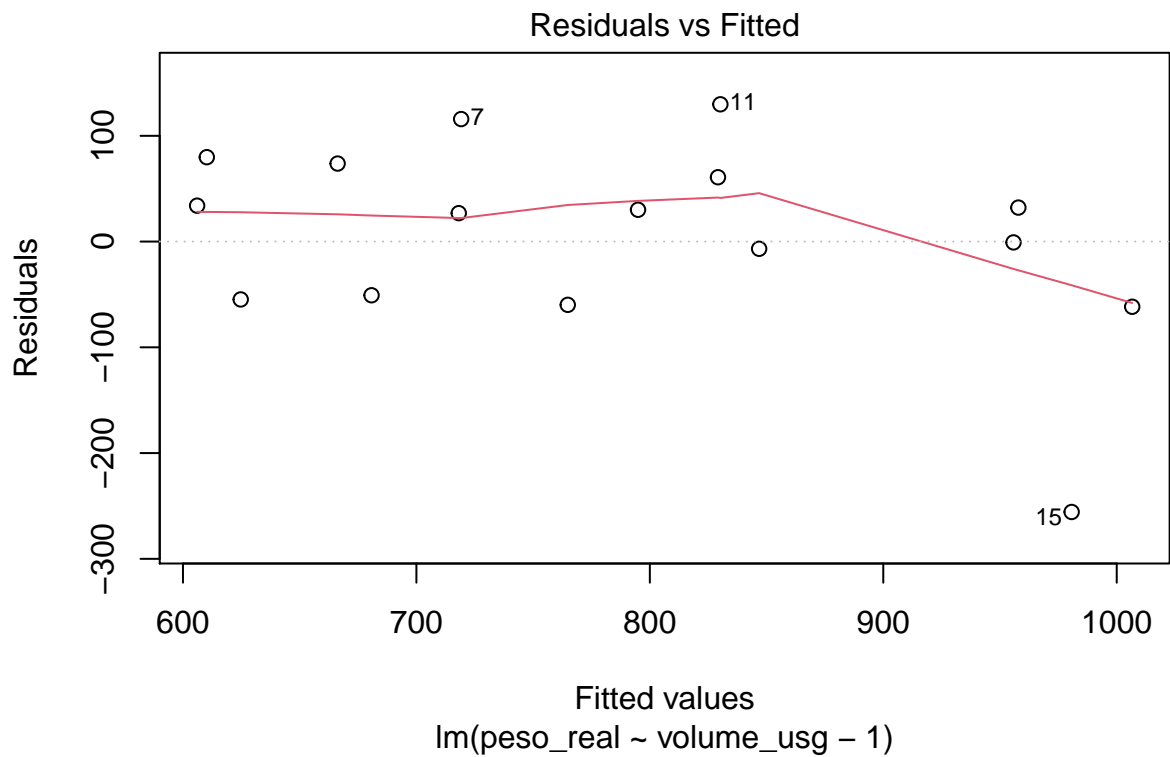
```
ajustarModelo <- function(dados) {  
  # - 1 omite o intercepto  
  ajuste <- lm(peso_real ~ volume_usg - 1, data=dados)  
  print(ajuste$coefficients)  
  intercept <- 0  
  slope <- ajuste$coefficients  
  p <- ggplot(dados, aes(x=volume_usg, y=peso_real)) + geom_point() + geom_abline(intercept = intercept, slope = slope)  
  plot(p)  
  print(summary(ajuste))  
  plot(ajuste)  
  
  return(ajuste)  
}  
  
ajuste <- ajustarModelo(dados3)
```

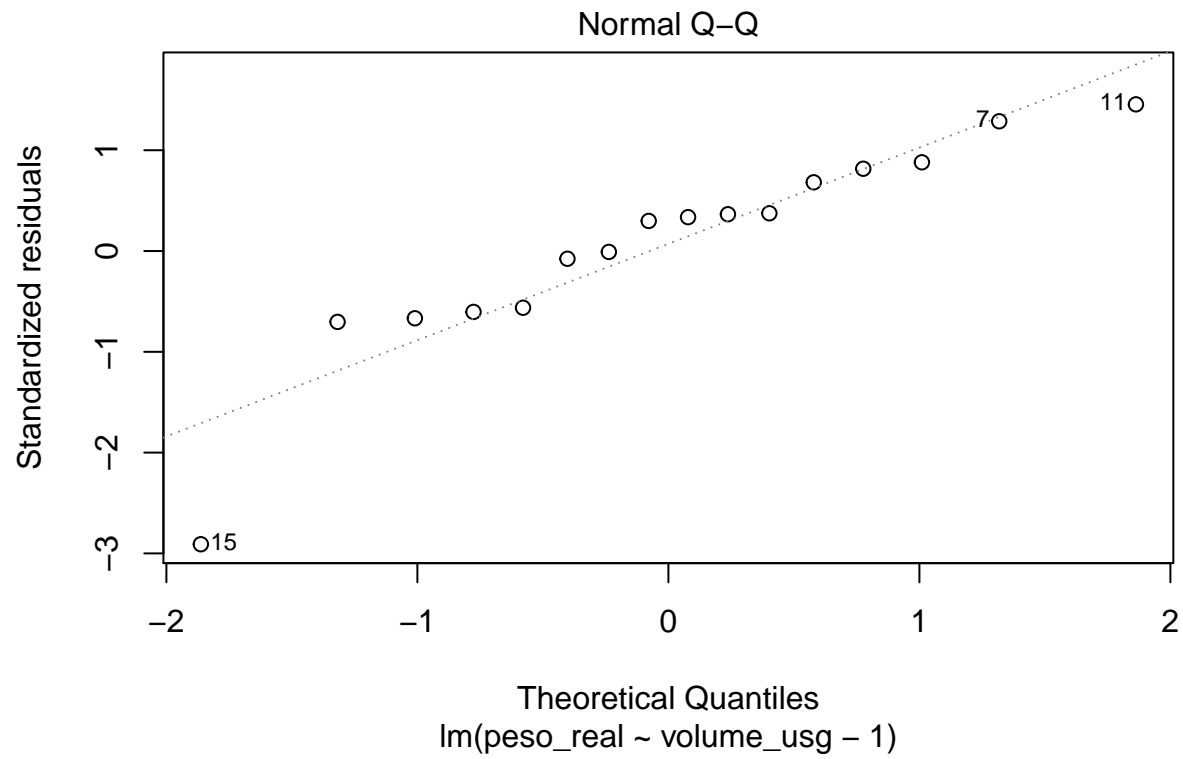
```
## volume_usg  
##      1.03777
```

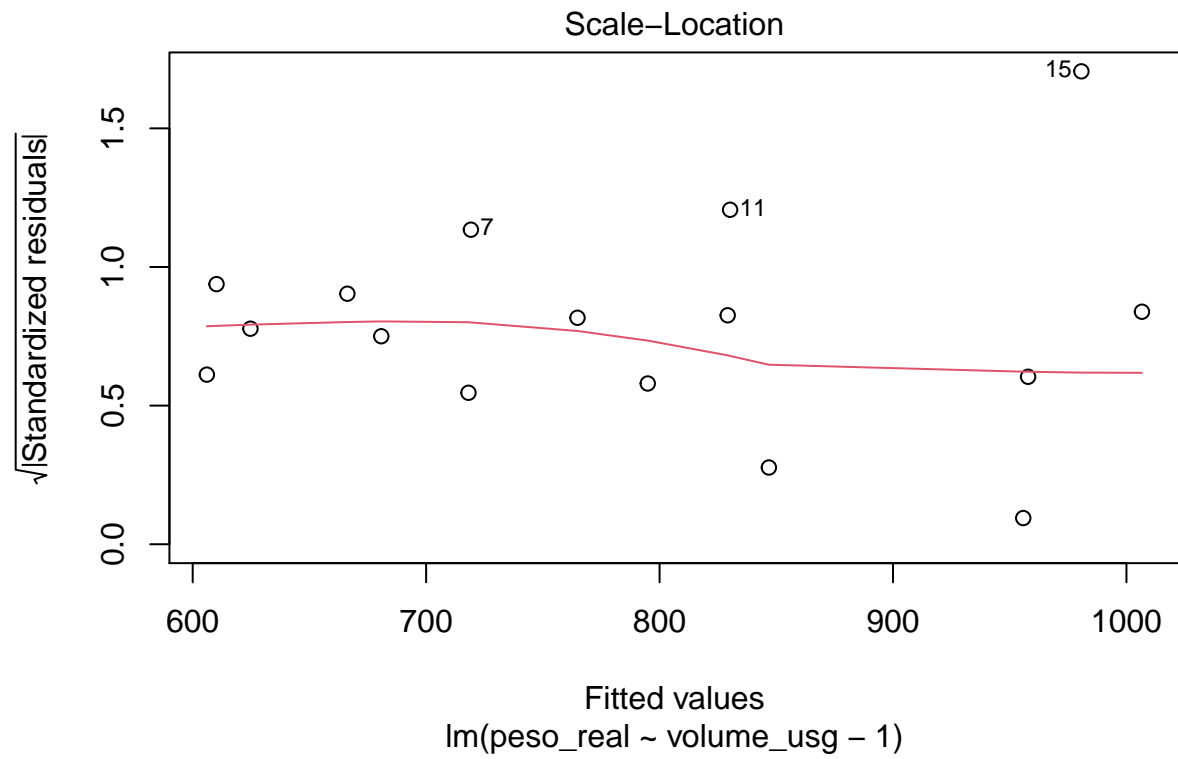


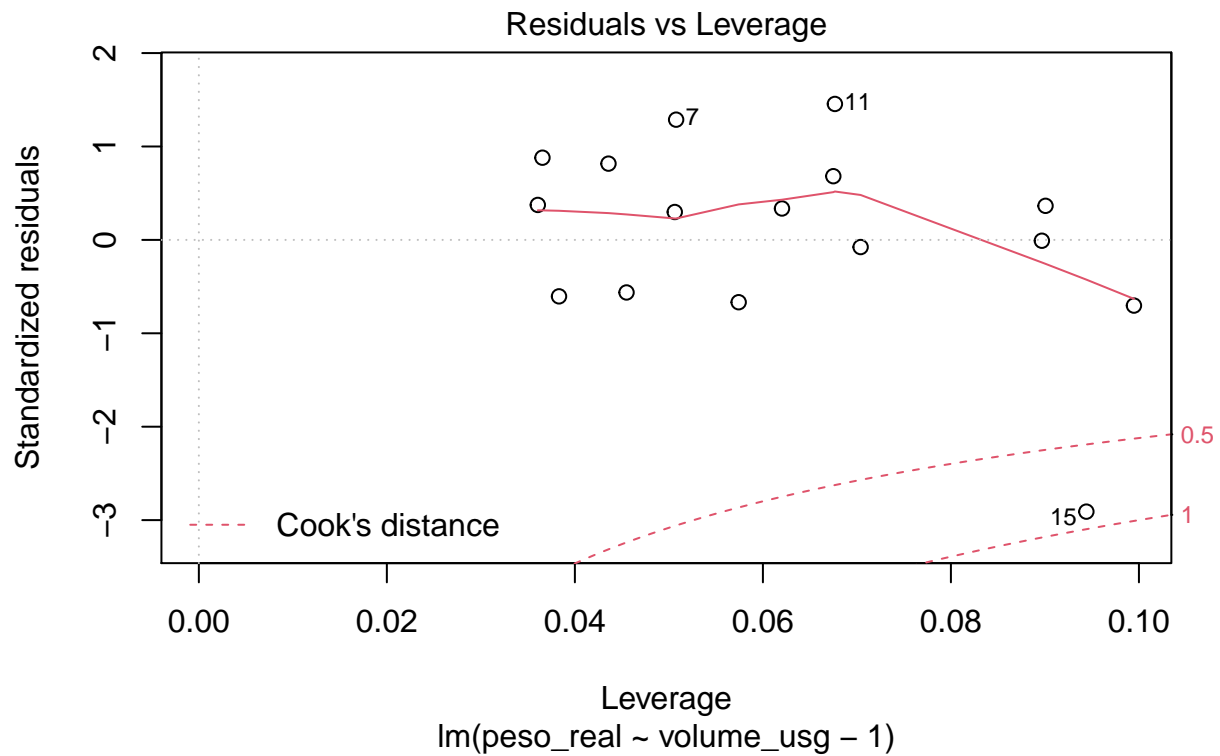
```
##
```

```
## Call:
## lm(formula = peso_real ~ volume_usg - 1, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -255.69  -51.77   28.47   64.06  129.78
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## volume_usg  1.03777     0.03003   34.56 1.03e-15 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 92.36 on 15 degrees of freedom
## Multiple R-squared:  0.9876, Adjusted R-squared:  0.9868
## F-statistic: 1194 on 1 and 15 DF, p-value: 1.026e-15
```





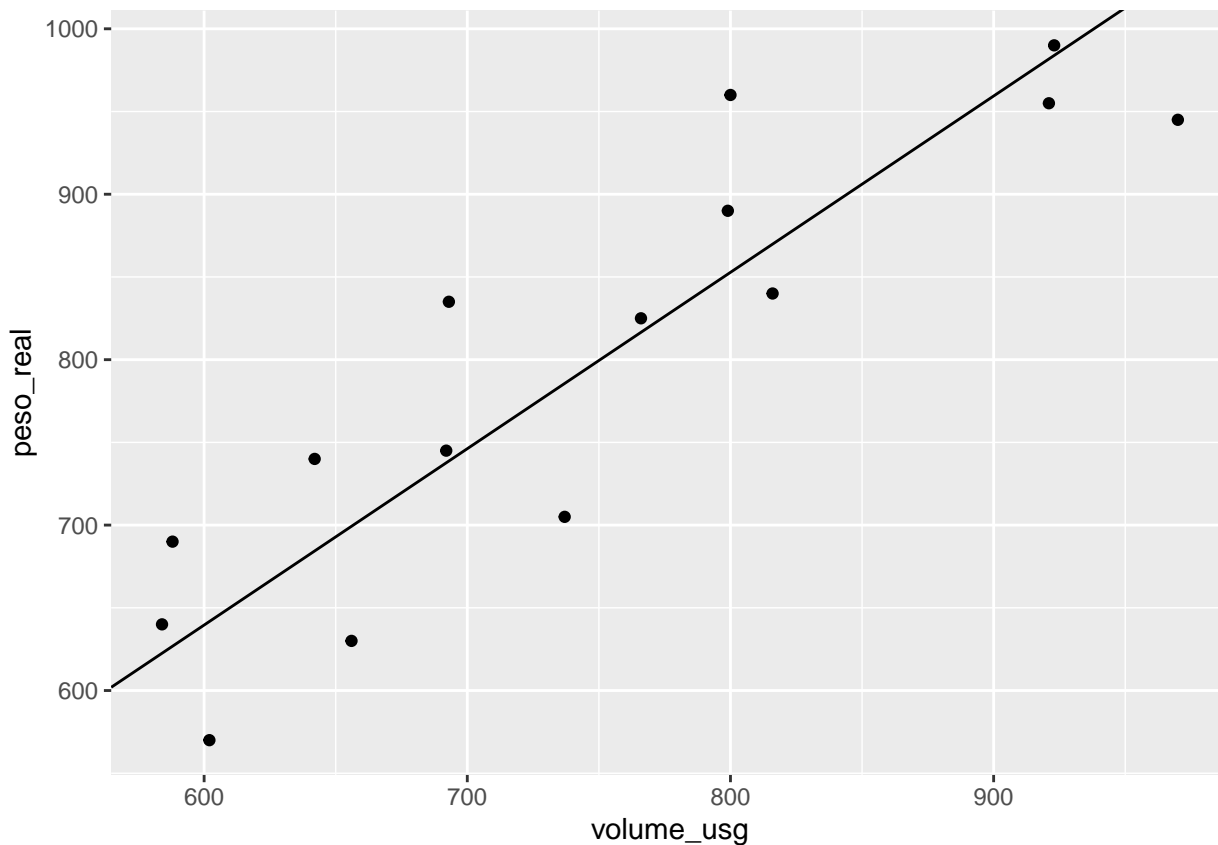




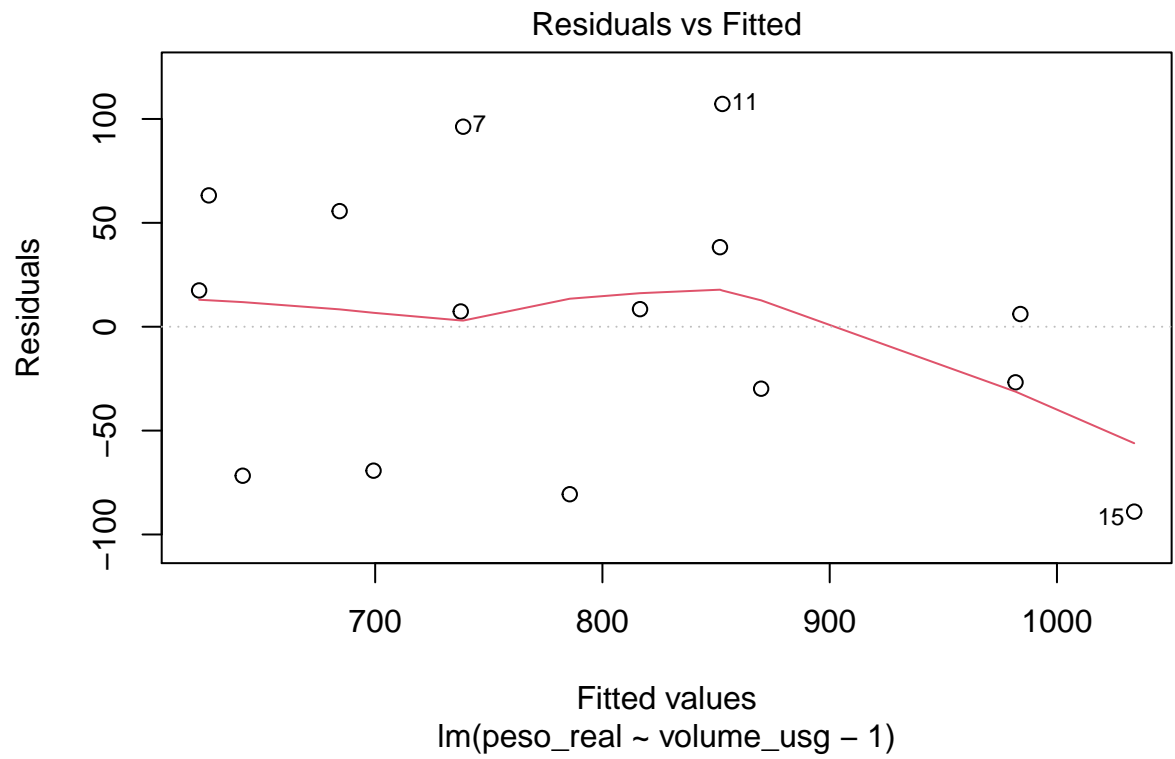
Novamente, os gráficos indicam que a observação 15 é um outlier. Refaremos o ajuste removendo a observação 15.

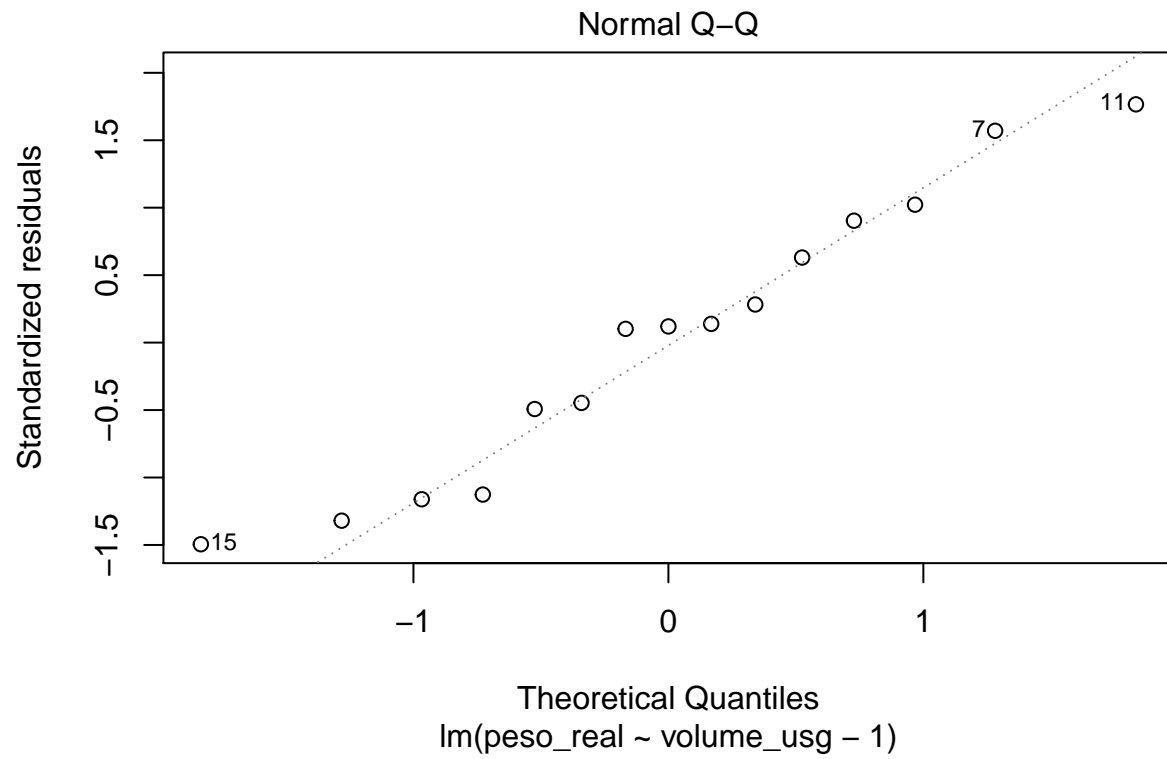
```
dados4 <- dados1[-c(15), ]
ajuste <- ajustarModelo(dados4)
```

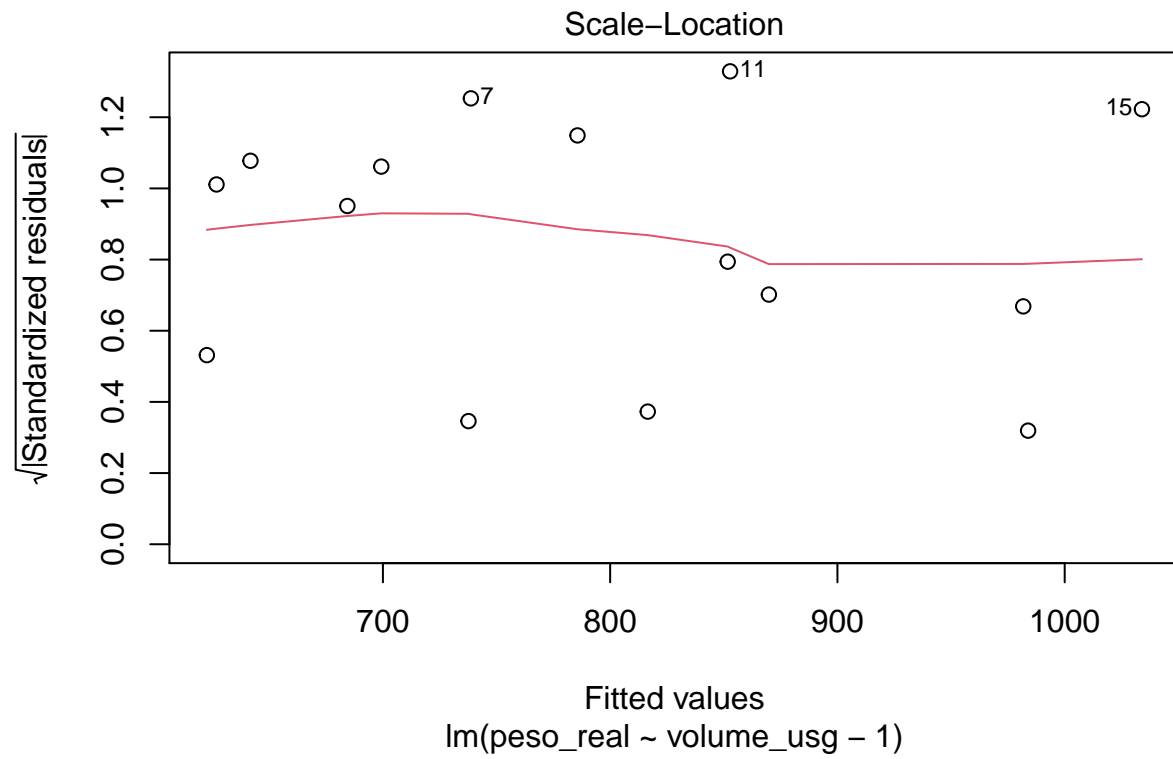
```
## volume_usg
## 1.065977
```

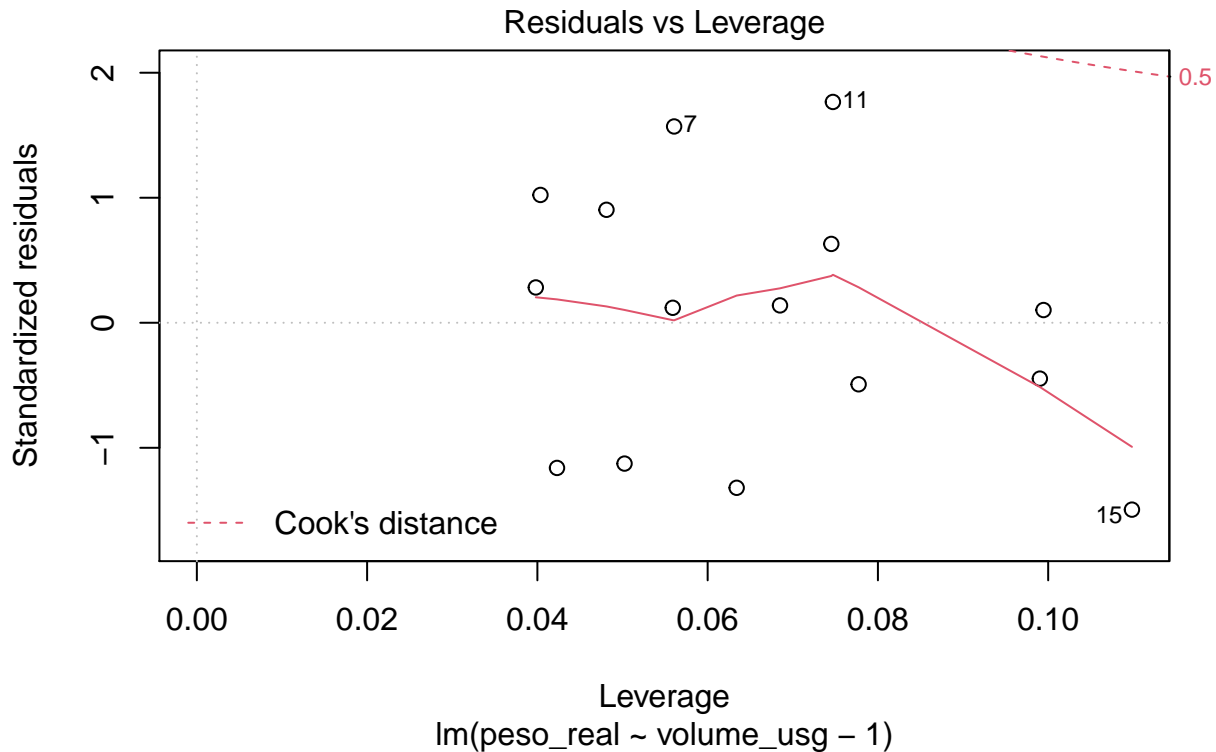


```
##
## Call:
## lm(formula = peso_real ~ volume_usg - 1, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -88.998 -49.559   7.344  46.963 107.218
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## volume_usg  1.06598    0.02156   49.44  <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 63.11 on 14 degrees of freedom
## Multiple R-squared:  0.9943, Adjusted R-squared:  0.9939
## F-statistic: 2444 on 1 and 14 DF, p-value: < 2.2e-16
```









As mesmas observações sobre a qualidade do modelo se aplicam. Os gráficos indicam que os resíduos possuem os valores dentro do esperado. Idealmente, o R^2 deveria estar próximo de 1, mas não está. Dessa forma, podemos concluir que o ajuste do modelo aproxima os dados, mas não estritamente. Assim, espera-se que o intervalo de confiança seja grande.

vi)iv)

Construindo intervalos de confiança dos parâmetros:

```
confint(ajuste)
```

```
##           2.5 %   97.5 %
## volume_usg 1.019729 1.112225
```

vi)v)

A seguir, construiremos a tabela.

```
volumes <- c(600, 700, 800, 900, 1000)
df <- data.frame(volume_usg = volumes)
previsto <- predict(ajuste, df, interval='confidence')
previsto <- data.frame(previsto)
intervalo <- previsto$fit - previsto$lwr
previsto <- cbind(volume_usg = volumes, previsto, intervalo = intervalo)
kable(previsto)
```

volume_usg	fit	lwr	upr	intervalo
600	639,59	611,84	667,34	27,75
700	746,18	713,81	778,56	32,37
800	852,78	815,78	889,78	37,00
900	959,38	917,76	1.001,00	41,62
1.000	1.065,98	1.019,73	1.112,23	46,25

vi)vi) Ambos os modelos satisfazem de forma similar as métricas mostradas na etapa (iii). Entretanto, observa-se na etapa (v) que o segundo modelo apresenta intervalos de confiança menores para suas predições de peso real. Dessa forma, o modelo sem intersecto demonstrou-se mais conveniente. Destacamos que o intervalo de confiança de 97,5% do parâmetro α no primeiro modelo era consideravelmente alto, o que poderia indicar que ele não possuía muita importância no modelo.

Exercício 2

Exercício 3

Exercício 4

Exercício 15

Exercício 16