Regressão Linear

Métodos Quantitativos Aplicados à Ciência Política

Frederico Bertholini

14.dez.2020

1 Modelos Lineares Multivariados

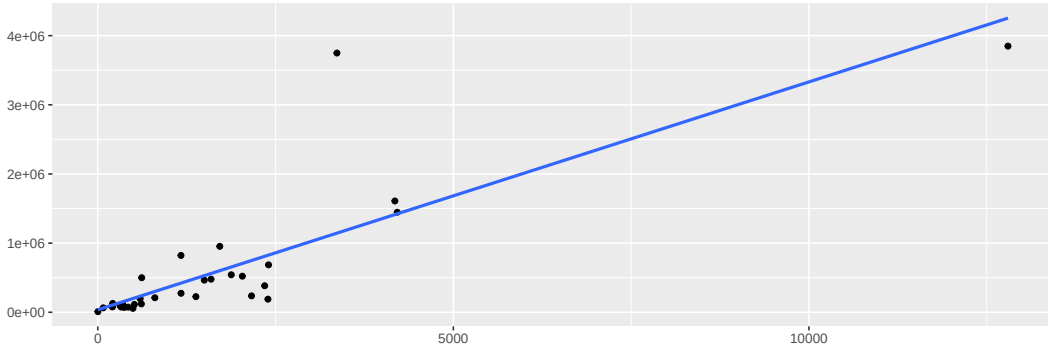
Conceitos

Modelos de regressão estabelecem relações entre variáveis.

Isso é feito através de uma equação que expressa uma variável **dependente** em termos de uma ou mais variáveis **independentes**.

Visualizando relações entre variáveis

```
legenda %>% ggplot(aes(x = DOAPFIS, y = VOTLEG)) +
  geom_point() + # Adiciona os pontos
  geom_smooth(method = "lm",se=F) + # Adiciona a curva, estimada por um modelo linear
  labs(x="",y="")
```



Tendência geral: $VOTLEG = \beta_0 + \beta_1 DOAPFIS$

4 perguntas importantes:

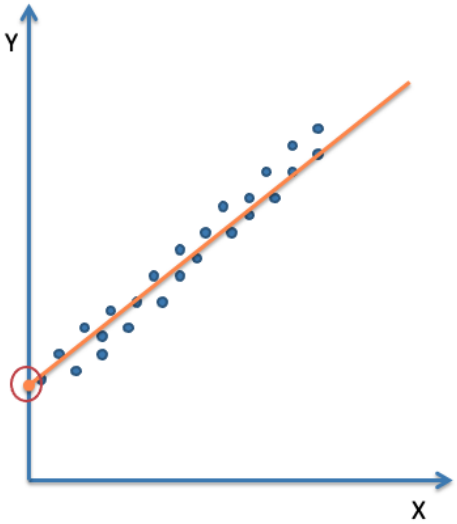
Quem é a varável dependente?

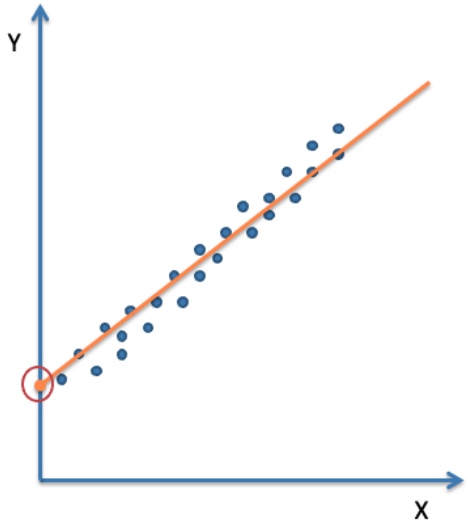
Quem é a independente?

Qual é o significado de β_1?

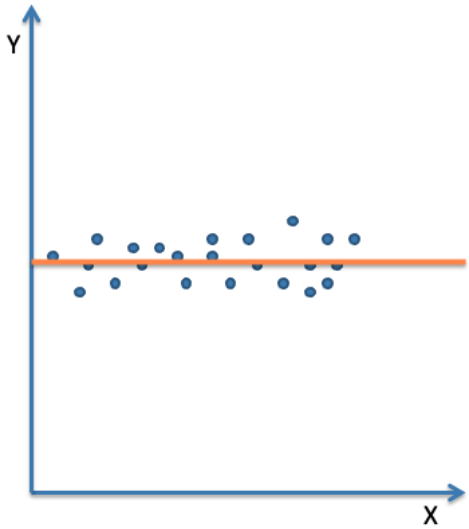
Qual é o significado de β_0?

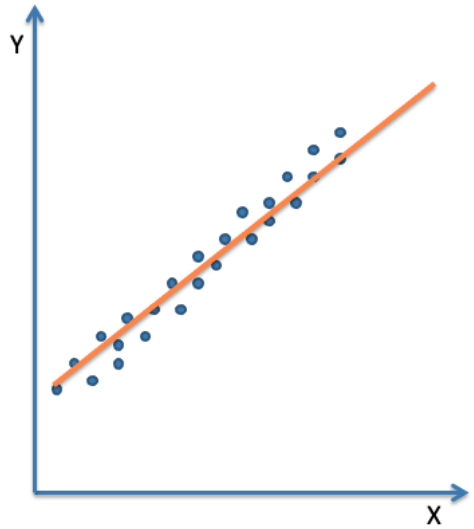
β_0





β_1





X e Y estão relacionadas. Esta relação ocorre para todos os X´s e Y´s.

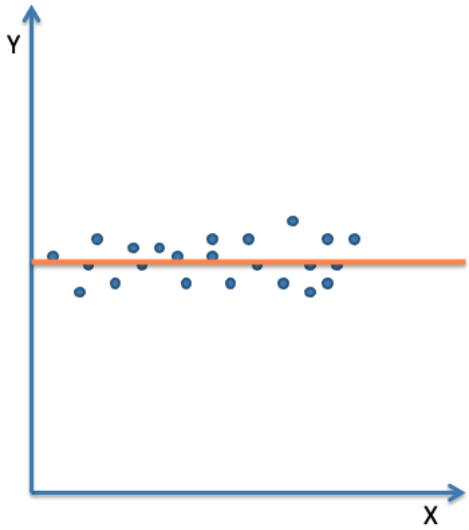
Nós coletamos alguns dados e possuímos apenas uma amostra de toda a população de X e Y. Observando a relação entre os X's e Y's da nossa amostra, nós tentamos estimar a relação entre X e Y na população.

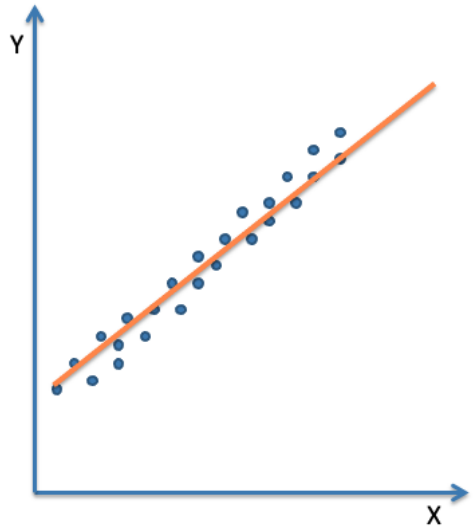
$$Y_i = \beta_0 + \beta_1 x_{i1} + \epsilon_i$$

$$Y_i = \hat{\beta}_0 + \hat{\beta}_1 X_i + \hat{\varepsilon}_i$$

$$\hat{Y} = b_0 + b_1 X$$

O que estamos testando? (e se $\beta_1 = 0$)





Significância e valor-p da regressão

$$\begin{cases} H_0 : \beta_1 = 0 \\ H_A : \beta_1 \neq 0 \end{cases}$$

Teste da significância de β_1:

Valor-p $<$ o que considerarmos adequado (0,05?) : Rejeita H_0

-> A relação entre X e Y é **significante**, ou seja, tem significância do ponto de vista estatístico.

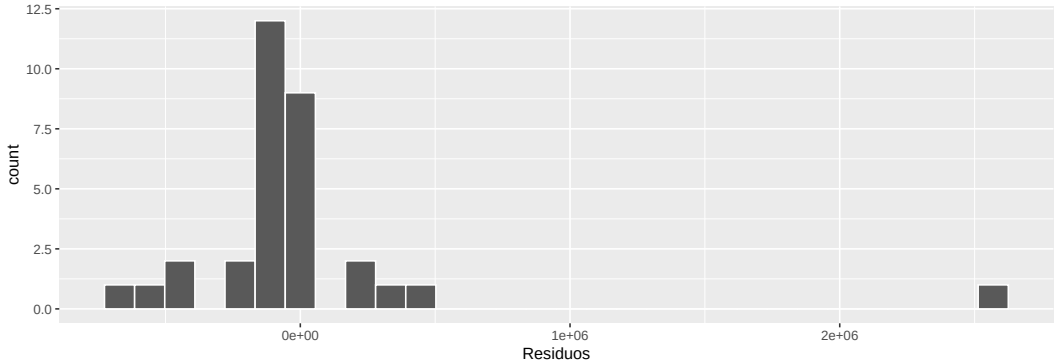
ps.: Significante é o mesmo que significativo?

Estimativas

```
(meu_modelo <- legenda %>% lm(VOTLEG ~ DOAPFIS, data = .))
```

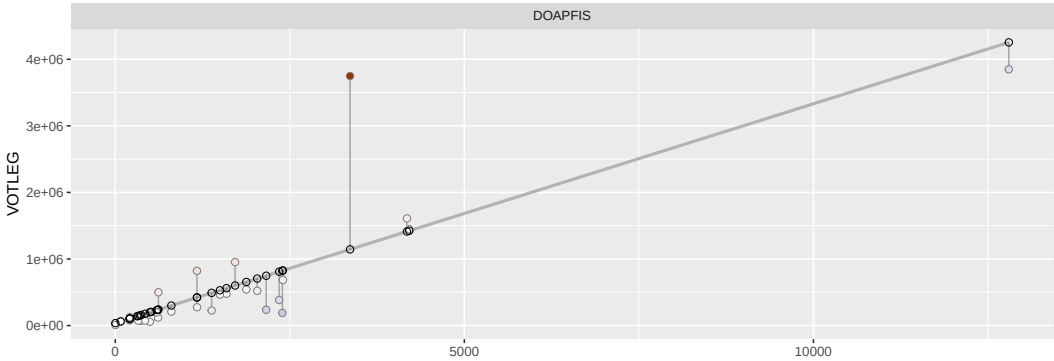
```
Call:
lm(formula = VOTLEG ~ DOAPFIS, data = .)

Coefficients:
(Intercept)      DOAPFIS
    36142.3        329.5
```

```
get_regression_points(meu_modelo) %>%
  ggplot(aes(x = residual)) +
  geom_histogram(color = "white") +
  labs(x = "Residuos")
```



Resíduos com `sjPlot`

```
sjPlot::plot_residuals(meu_modelo)
```



Quais são as características ideais da nossa reta?

Resíduo esperado é zero: $E(e) = 0$

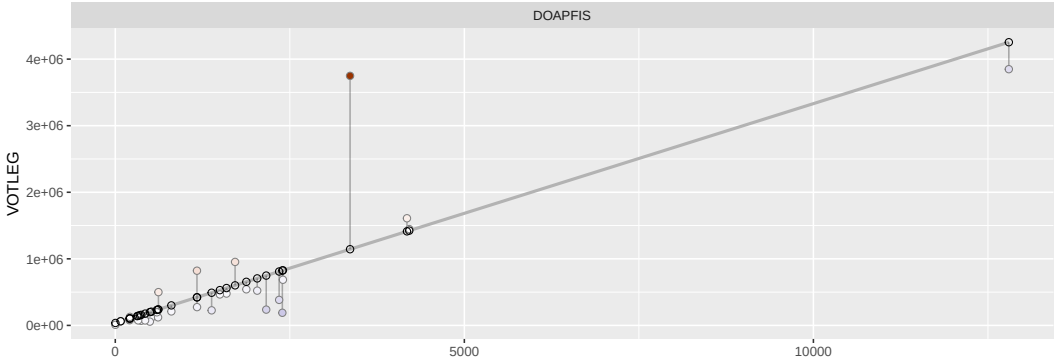
Erra igualmente para ambos os lados

Erra o mínimo possível

Método dos mínimos quadrados

Escolhe uma reta de forma que a soma dos erros ao quadrado seja a menor possível.

Encontrar os valores de b_0 e b_1 para o qual é o menor possível.



MMQ ou OLS ...

Encontrar os valores de b_0 e b_1 que minimizem

$$S = \sum_i [Y_i - (b_0 + b_1 X)]^2$$

Obtemos isso calculando b_0 e b_1 tais que

$$\frac{\partial S}{\partial b_0} = 0 \quad \frac{\partial S}{\partial b_1} = 0$$

Resolvendo, chega-se a:

$$b_1 = \frac{n \sum_i X_i Y_i - \left(\sum_i X_i\right)\left(\sum_i Y_i\right)}{n \sum_i X_i^2 - \left(\sum_i X_i\right)^2}$$

$$b_0 = \frac{\left(\sum_i Y_i\right)\left(\sum_i X_i^2\right) - \left(\sum_i X_i\right)\left(\sum_i X_i Y_i\right)}{n \sum_i X_i^2 - \left(\sum_i X_i\right)^2}$$

. . . continua

Que pode ser escrito como

$$b_1 = \frac{\text{cov}(X, Y)}{\text{var}(X)}$$

$$\bar{Y} = b_0 + b_1 \bar{X}$$

Sob certas premissas, é possível provar que

$$\frac{b_1}{s_{b_1}} \sim t_{n-2}$$

Onde

$$s_{b_1}^2 = \frac{s^2}{\sum_i \left(X_i - \bar{X}\right)^2} \quad s_e^2 = \frac{\sum_i e_i^2}{n-2}$$

Isto permite:

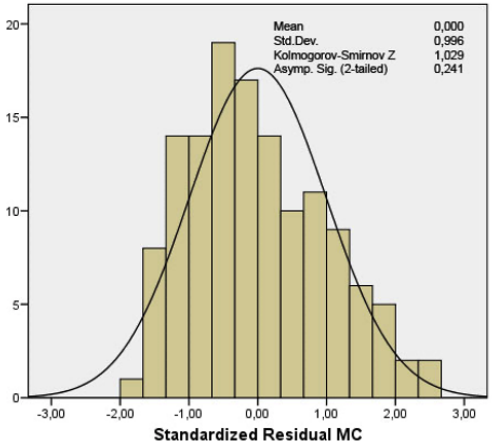
– realizar teste de hipóteses com b1, como, por exemplo, testar sua significância

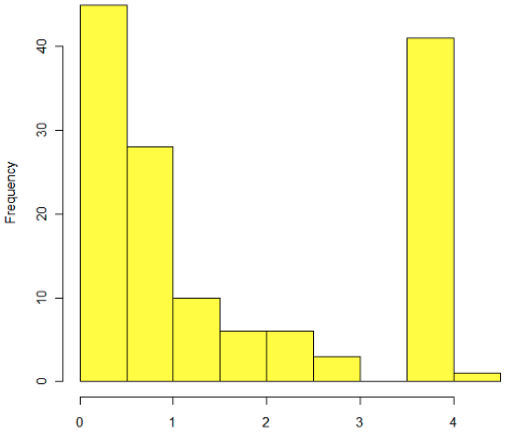
– Construir intervalos de confiança e de predição para Y em um valor de X qualquer

Premissas

Os resíduos devem ser normais, homoscedásticos e independentes.

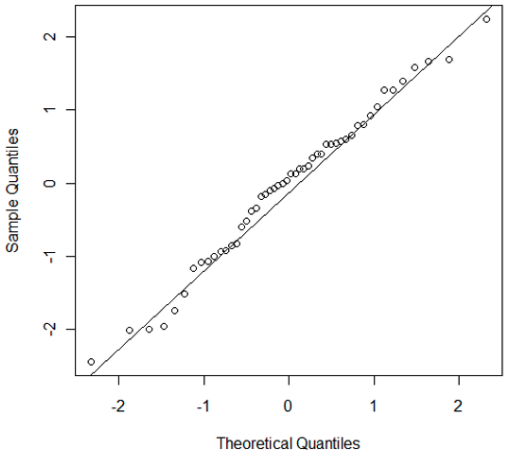
Normalidade



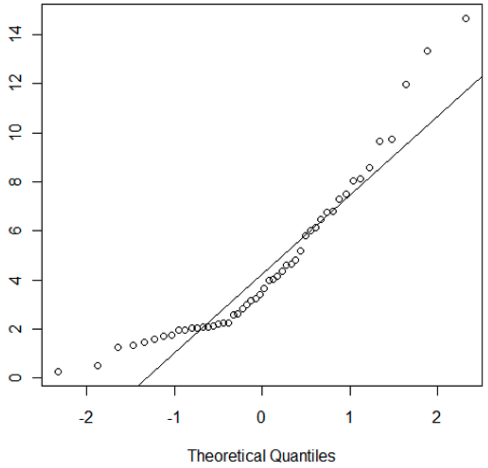


Normalidade

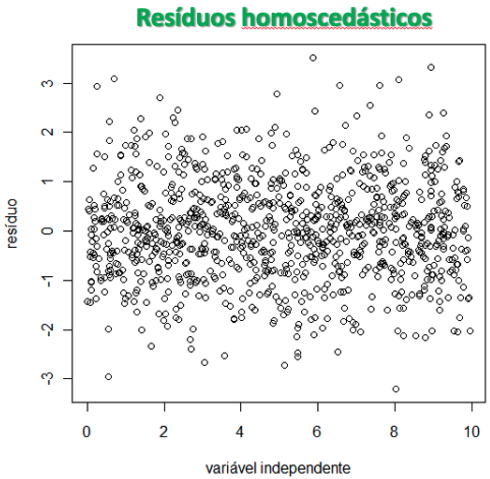
Resíduos normais

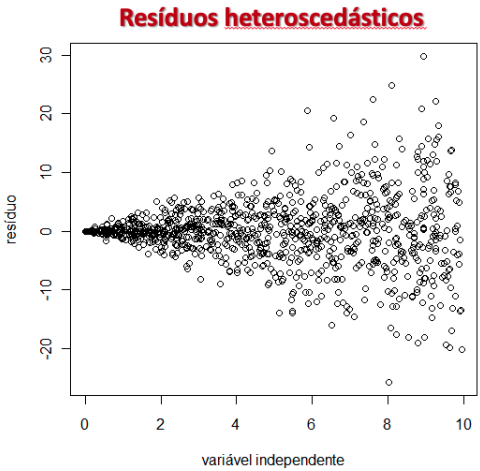


Resíduos não-normais



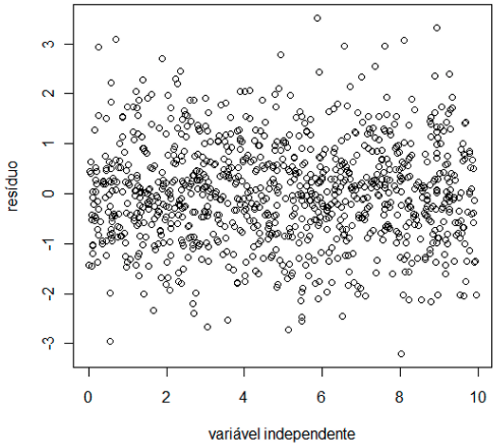
Homocedasticidade



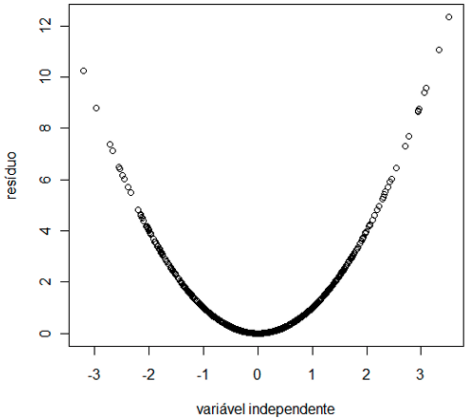


Independência

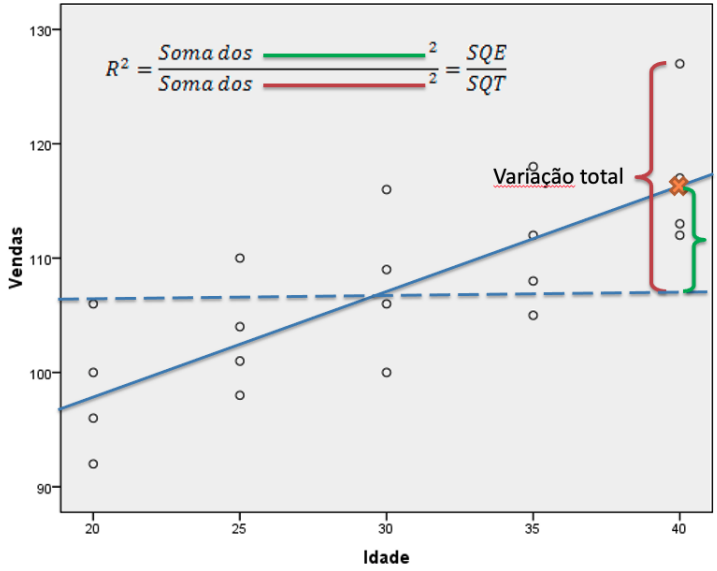
Resíduos independentes



Padrão de formação de resíduos



R^2



Variação explicada pelo modelo (Regression, Model, etc)

OU R^2= 1-[SQR/SQT]
É equivalente

Modelos explicativos e preditivos

Modelos de regressão podem servir para explicar ou para prever.

- Em modelos explicativos, o que importa é ter fortes razões para crer que as variáveis explicativas influenciam a variável explicada. Isso é medido por um valor-p baixo.
- Em modelos preditivos, o que importa é ter um bom ajuste da reta aos dados. Isso é medido por um R^2 alto.

Modelo preditivo: Previsão eleitoral (electoral forecasting)

Modelo explicativo: total de votos de legenda para deputado federal (VOTLEG) **explicado por** número de doações de pessoas físicas (DOAPFIS)

Eliminando outliers

```
legenda %<>% dplyr::filter(DOAPFIS<5000,VOTLEG<2000000)
(meu_modelo <- legenda %>% lm(VOTLEG ~ DOAPFIS, data = .))
```

```
Call:
lm(formula = VOTLEG ~ DOAPFIS, data = .)

Coefficients:
(Intercept)      DOAPFIS
    -6439.4        304.1
```

Usando `tidy`

```
tidy(meu_modelo)
```

```
# A tibble: 2 x 5
  term        estimate std.error statistic       p.value
  <chr>          <dbl>     <dbl>     <dbl>         <dbl>
1 (Intercept)   -6439.   58572.     -0.110 0.913
2 DOAPFIS         304.      35.9     8.47  0.00000000325
```

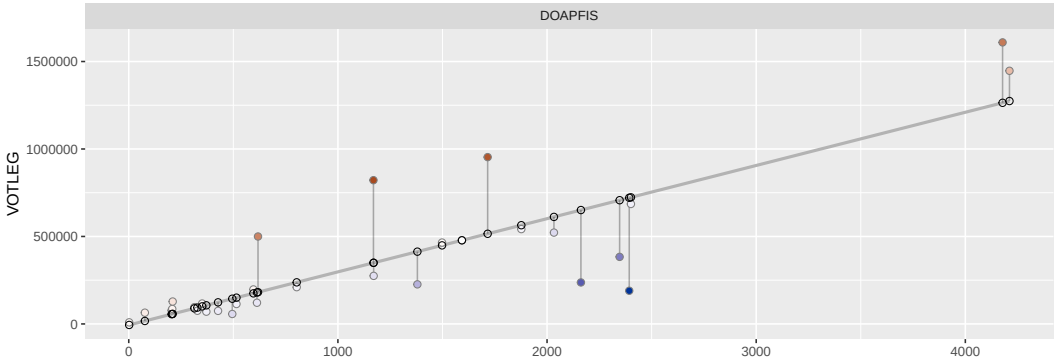
Usando `glance`

```
glance(meu_modelo)
```

```
# A tibble: 1 x 12
  r.squared adj.r.squared  sigma statistic p.value    df logLik   AIC   BIC
      <dbl>         <dbl>  <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
1     0.719         0.709 2.16e5      71.8 3.25e-9     1  -410.  826.  830.
# ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

Resíduos com `sjPlot`

```
sjPlot::plot_residuals(meu_modelo)
```



Inferência

NC = 95%

```
confint(meu_modelo, level = 0.95)
```

```
                  2.5 %      97.5 %
(Intercept) -126418.782 113539.9612
DOAPFIS         230.576    377.5736
```

NC = 90%

```
confint(meu_modelo, level = 0.90)
```

```
                   5 %      95 %
(Intercept) -106078.1080 93199.288
DOAPFIS         243.0366   365.113
```

É possível salvar o *output* da função `summary`

```
(resumo <- summary(meu_modelo))
```

```
Call:
lm(formula = VOTLEG ~ DOAPFIS, data = .)

Residuals:
    Min      1Q  Median      3Q     Max
-531447  -56292   -8653   27238  472375

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -6439.41   58572.03  -0.110    0.913
DOAPFIS       304.07      35.88   8.475 3.25e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 215600 on 28 degrees of freedom
Multiple R-squared:  0.7195,    Adjusted R-squared:  0.7095
F-statistic: 71.82 on 1 and 28 DF,  p-value: 3.249e-09
```

Obtendo resultados detalhados

```
meu_modelo$coefficients
```

```
(Intercept)     DOAPFIS
 -6439.4103    304.0748
```

Outras informações salvas dentro do objeto podem ser vistas com `names`:

```
names(meu_modelo)
```

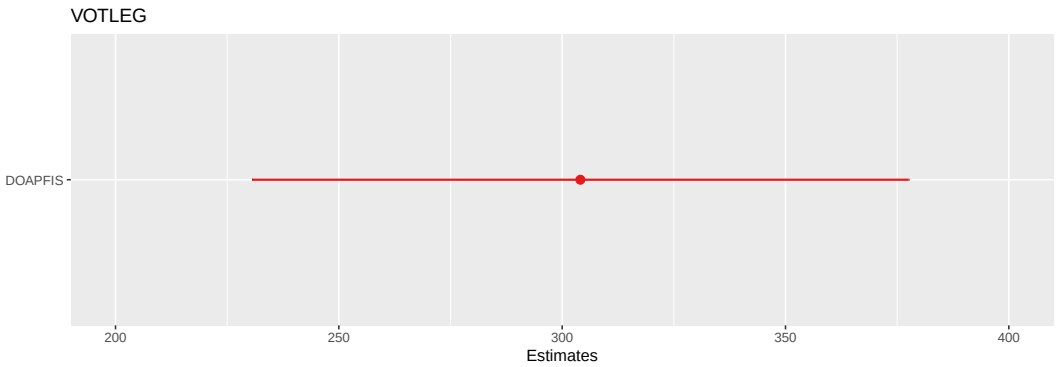
```
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "xlevels"       "call"          "terms"         "model"
```

R^2

```
resumo$r.squared
```

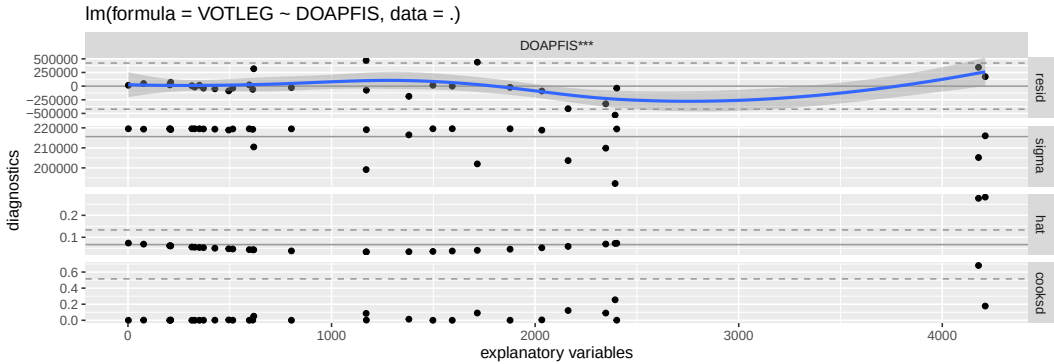
```
[1] 0.7194894
```

```
sjPlot::plot_model(meu_modelo)
```

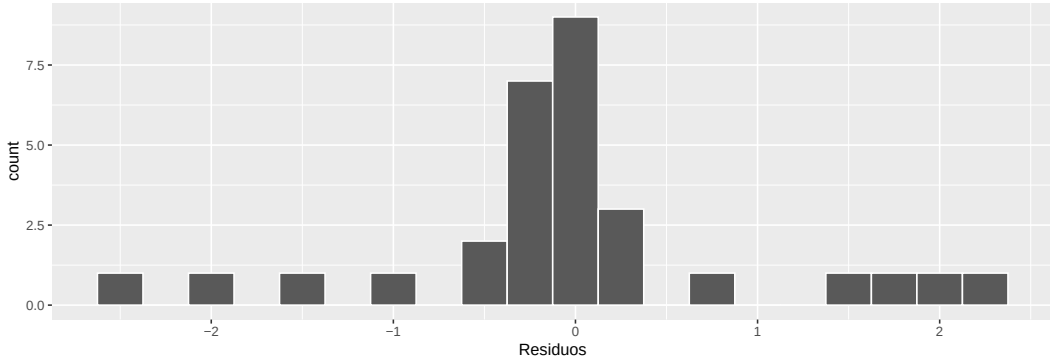


Diagnósticos

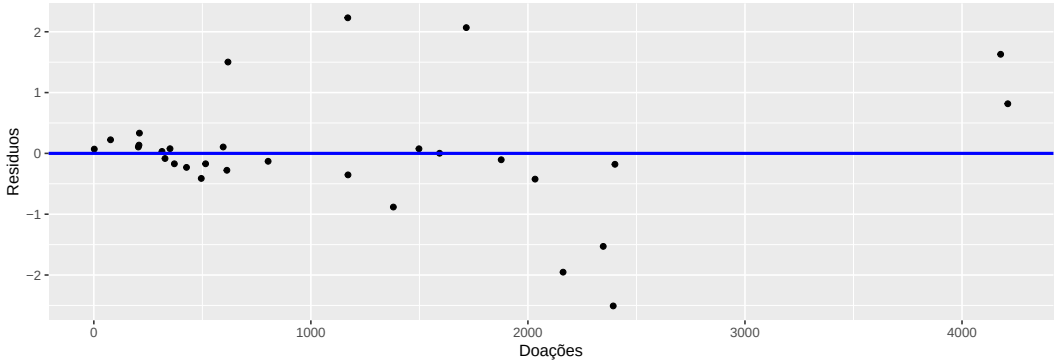

```
GGally::ggnostic(meu_modelo)
```



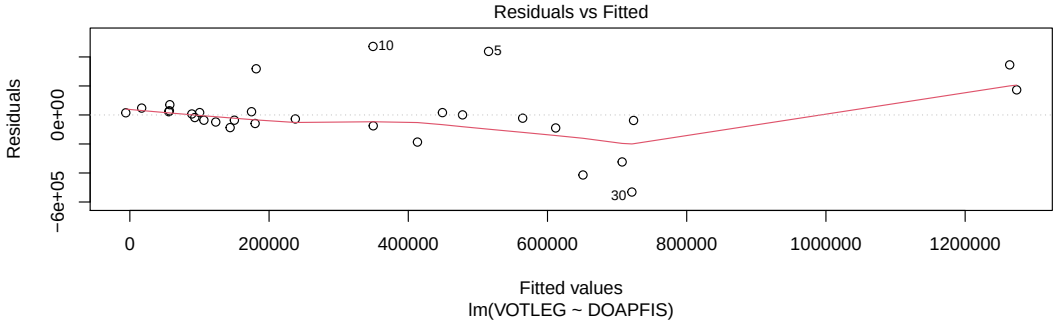
```
get_regression_points(meu_modelo) %>% mutate(residual=scale(residual)) %>%
  ggplot(aes(x = residual)) +
  geom_histogram(binwidth = .25,color = "white") +
  labs(x = "Residuos")
```



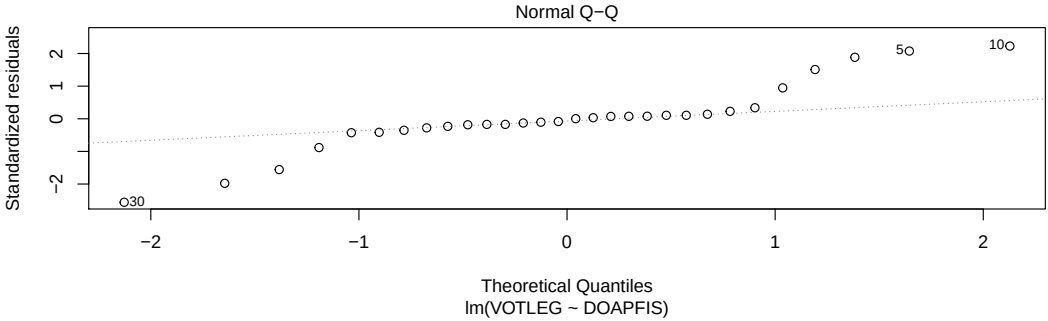
```
get_regression_points(meu_modelo) %>% mutate(residual=scale(residual)) %>%
  ggplot(aes(x = DOAPFIS, y = residual)) +
  geom_point() + labs(x = "Doações", y = "Residuos") +
  geom_hline(yintercept = 0, col = "blue", size = 1)
```



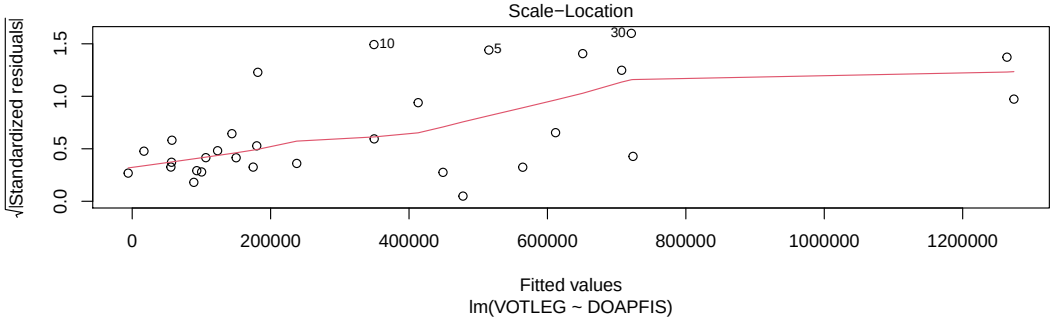
```
plot(meu_modelo,1)
```



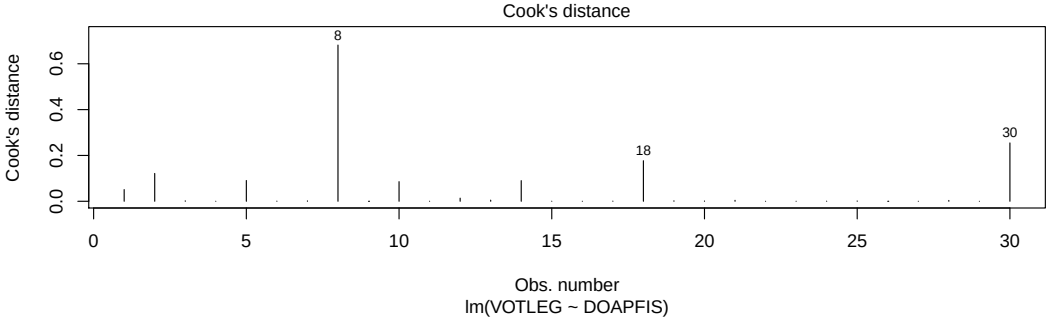
```
plot(meu_modelo,2)
```



```
plot(meu_modelo,3)
```



```
plot(meu_modelo,4)
```



Modelos Lineares Multivariados

Modelo linear com dois preditores

```
(meu_modelo2 <- legenda %>% lm(VOTLEG ~ DOAPFIS + NUMCAND, data = .))
```

```
Call:
lm(formula = VOTLEG ~ DOAPFIS + NUMCAND, data = .)

Coefficients:
(Intercept)      DOAPFIS      NUMCAND
  -148413.4        224.1       1685.9
```

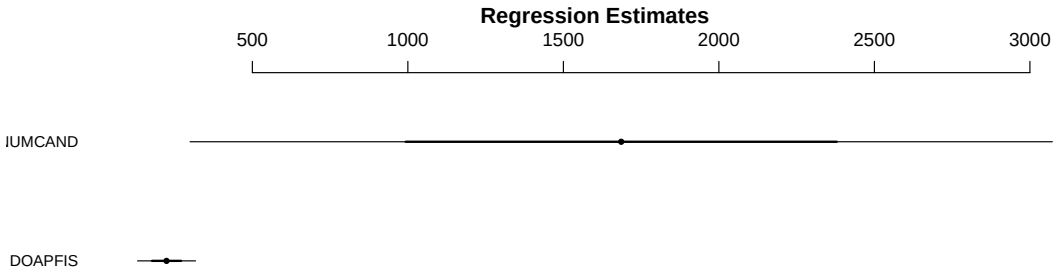
Obtendo resultados simplificados com `arm`

```
display(meu_modelo2)
```

```
lm(formula = VOTLEG ~ DOAPFIS + NUMCAND, data = .)
            coef.est   coef.se
(Intercept) -148413.43   79529.76
DOAPFIS         224.12      46.64
NUMCAND        1685.94     693.11
---
n = 30, k = 3
residual sd = 198855.86, R-Squared = 0.77
```

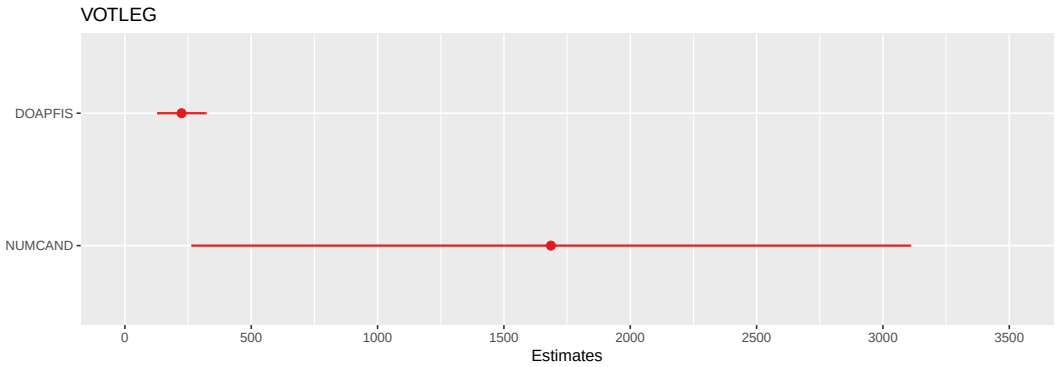
Interpretando resultados com gráficos


```
coefplot(meu_modelo2)
```



Interpretando resultados com gráficos


```
sjPlot::plot_model(meu_modelo2)
```



```
sjPlot::plot_residuals(meu_modelo2)
```