Regressão - Aplicações

Professor João Gabriel de Moraes Souza

29/07/2022

Esse exemplo segue baseado na aplicação feita em "LAMFO Aplicações em Regressão" pelo assistente de pesquisa do *LAMFO* João Pedro Fontoura da Silva.

Regressão Linear Simples

Como exemplo de aplicação de regressão linear, queremos relacionar notas de testes com a proporção de estudantes por professor obtidos de uma base de dados referentes a escolas da Califórnia (EUA). A nota dos testes (**TestScore**) é a média das notas de leitura e matemática para classes do 5º ano; já o tamanho das salas é medido pela proporção de estudantes relativa à quantidade de professores (que a partir deste ponto será identificada como *STR*, ou student-teacher ratio). Os dados são provenientes do banco de dados CASchools, contido no pacote AER disponível para R.

Importando as Bibliotecas Necessárias

```
suppressMessages(library(AER))
suppressMessages(library(ggplot2))
suppressMessages(library(ggpubr))
suppressMessages(library(olsrr))
suppressMessages(library(car))
suppressMessages(library(sandwich))
data(CASchools)
head(CASchools)
```

```
##   district                          school  county grades students teachers
## 1    75119              Sunol Glen Unified Alameda  KK-08      195    10.90
## 2    61499            Manzanita Elementary   Butte  KK-08      240    11.15
## 3    61549     Thermalito Union Elementary   Butte  KK-08     1550    82.90
## 4    61457 Golden Feather Union Elementary   Butte  KK-08      243    14.00
## 5    61523        Palermo Union Elementary   Butte  KK-08     1335    71.50
## 6    62042         Burrel Union Elementary  Fresno  KK-08      137     6.40
##   calworks   lunch computer expenditure    income   english  read  math
## 1   0.5102  2.0408       67    6384.911 22.690001  0.000000 691.6 690.0
## 2  15.4167 47.9167      101    5099.381  9.824000  4.583333 660.5 661.9
## 3  55.0323 76.3226      169    5501.955  8.978000 30.000002 636.3 650.9
## 4  36.4754 77.0492       85    7101.831  8.978000  0.000000 651.9 643.5
## 5  33.1086 78.4270      171    5235.988  9.080333 13.857677 641.8 639.9
## 6  12.3188 86.9565       25    5580.147 10.415000 12.408759 605.7 605.4
```

É importante perceber que as duas variáveis de interesse não estão incluídas no pacote, então faz-se necessário computá-las manualmente a partir dos dados contidos em **CASchools**.

Computando os Dados de Interesse

Com isso iremos construir as variáveis de interesse do nosso exemplo.

```
CASchools$STR = CASchools$students/CASchools$teachers
CASchools$score = (CASchools$read + CASchools$math)/2
head(CASchools)
```

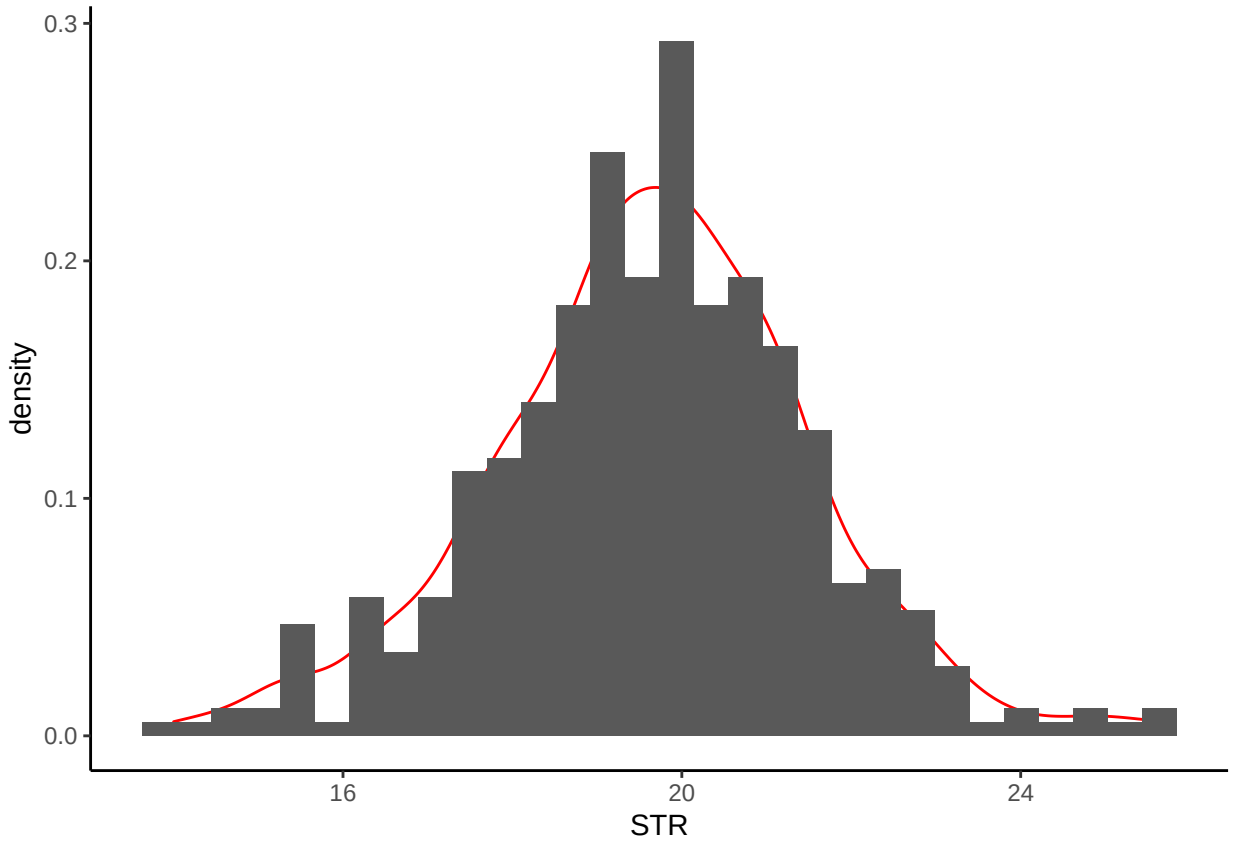
```
##   district                          school  county grades students teachers
## 1    75119              Sunol Glen Unified Alameda  KK-08      195    10.90
## 2    61499            Manzanita Elementary   Butte  KK-08      240    11.15
## 3    61549     Thermalito Union Elementary   Butte  KK-08     1550    82.90
## 4    61457 Golden Feather Union Elementary   Butte  KK-08      243    14.00
## 5    61523        Palermo Union Elementary   Butte  KK-08     1335    71.50
## 6    62042         Burrel Union Elementary  Fresno  KK-08      137     6.40
##   calworks   lunch computer expenditure    income   english  read  math
## 1   0.5102  2.0408       67    6384.911 22.690001  0.000000 691.6 690.0
## 2  15.4167 47.9167      101    5099.381  9.824000  4.583333 660.5 661.9
## 3  55.0323 76.3226      169    5501.955  8.978000 30.000002 636.3 650.9
## 4  36.4754 77.0492       85    7101.831  8.978000  0.000000 651.9 643.5
## 5  33.1086 78.4270      171    5235.988  9.080333 13.857677 641.8 639.9
## 6  12.3188 86.9565       25    5580.147 10.415000 12.408759 605.7 605.4
##        STR  score
## 1 17.88991 690.80
## 2 21.52466 661.20
## 3 18.69723 643.60
## 4 17.35714 647.70
## 5 18.67133 640.85
## 6 21.40625 605.55
```

Analisando as FDPs das variáveis de interesse

Plotando os gráficos para STR :

```
ggplot(CASchools) +
  geom_density(aes(x=STR), colour = "red") +
  geom_histogram(aes(x=STR, y=..density..)) +
  theme_classic()
```

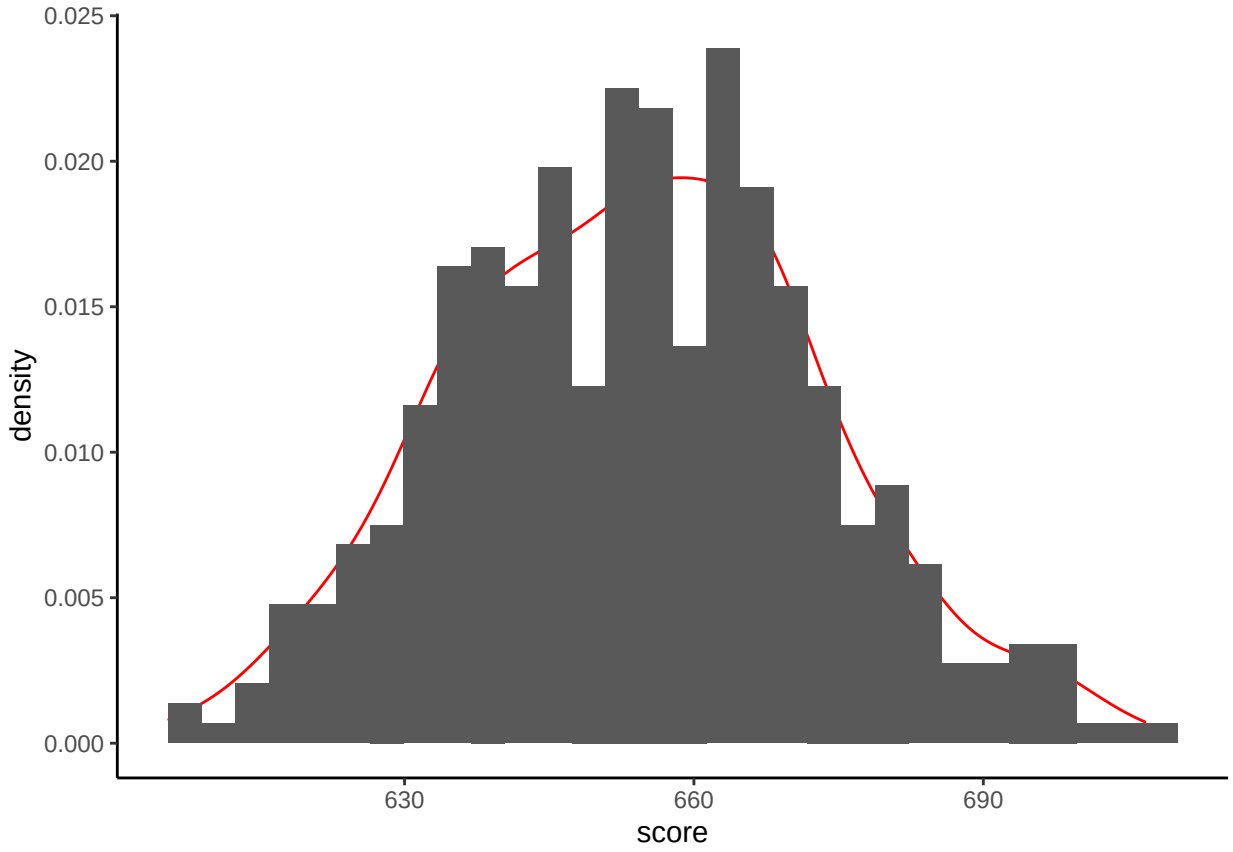
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Plotando os gráficos para *score* :

```
ggplot(CASchools) +
  geom_density(aes(x=score), colour = "red") +
  geom_histogram(aes(x=score, y=..density..)) +
  theme_classic()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Estimando o Modelo de Regressão

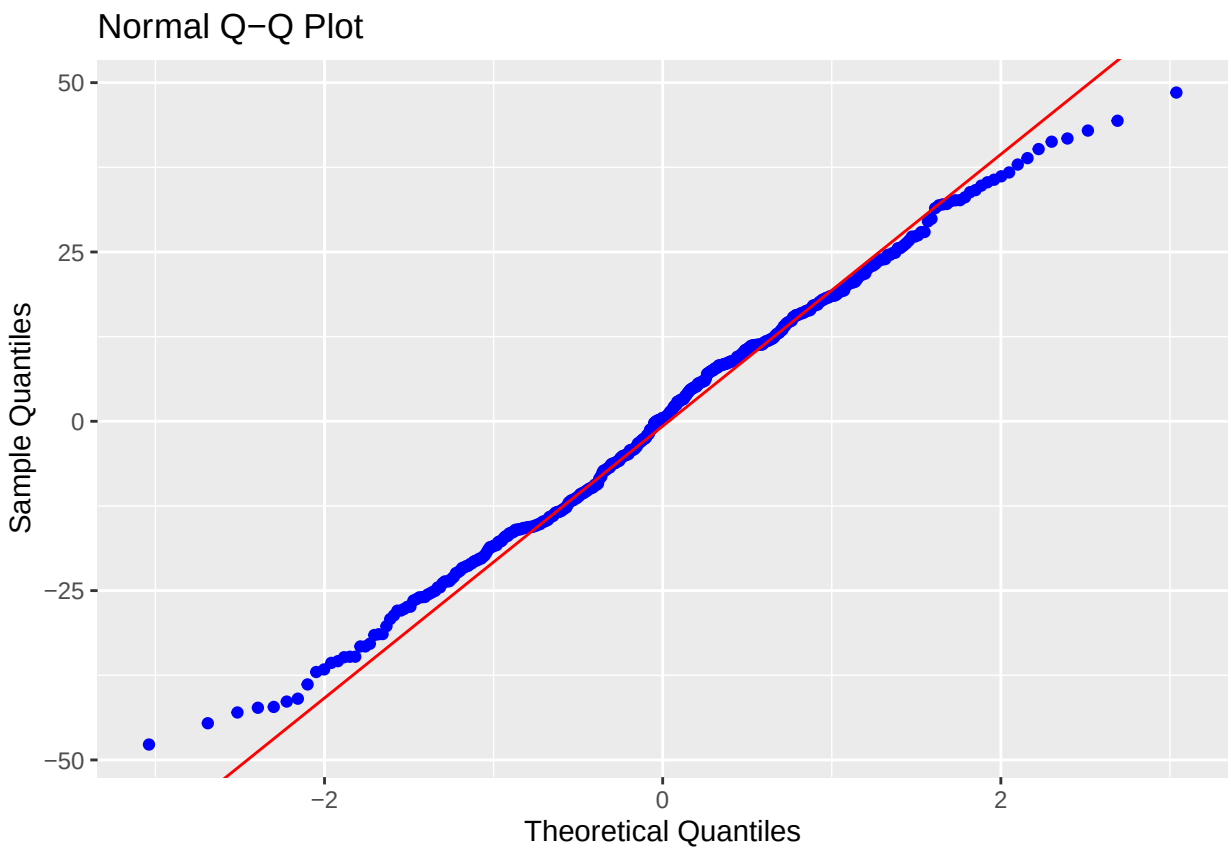
De modo a estimar o modelo por MQO, definindo **TestScore** como a variável dependente e **STR** como a variável independente, fazemos uso da função **lm()** do R para realizar uma regressão linear simples.

```
# Estimando o modelo
reg_linear <- lm(score ~ STR, data = CASchools)
summary(reg_linear)
```

```
##
## Call:
## lm(formula = score ~ STR, data = CASchools)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -47.727 -14.251   0.483  12.822  48.540
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 698.9329     9.4675  73.825  < 2e-16 ***
## STR          -2.2798     0.4798  -4.751 2.78e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.58 on 418 degrees of freedom
## Multiple R-squared:  0.05124,    Adjusted R-squared:  0.04897
## F-statistic: 22.58 on 1 and 418 DF,  p-value: 2.783e-06
```

Diagnóstico do Modelo

```
ols_plot_resid_qq(reg_linear)
```



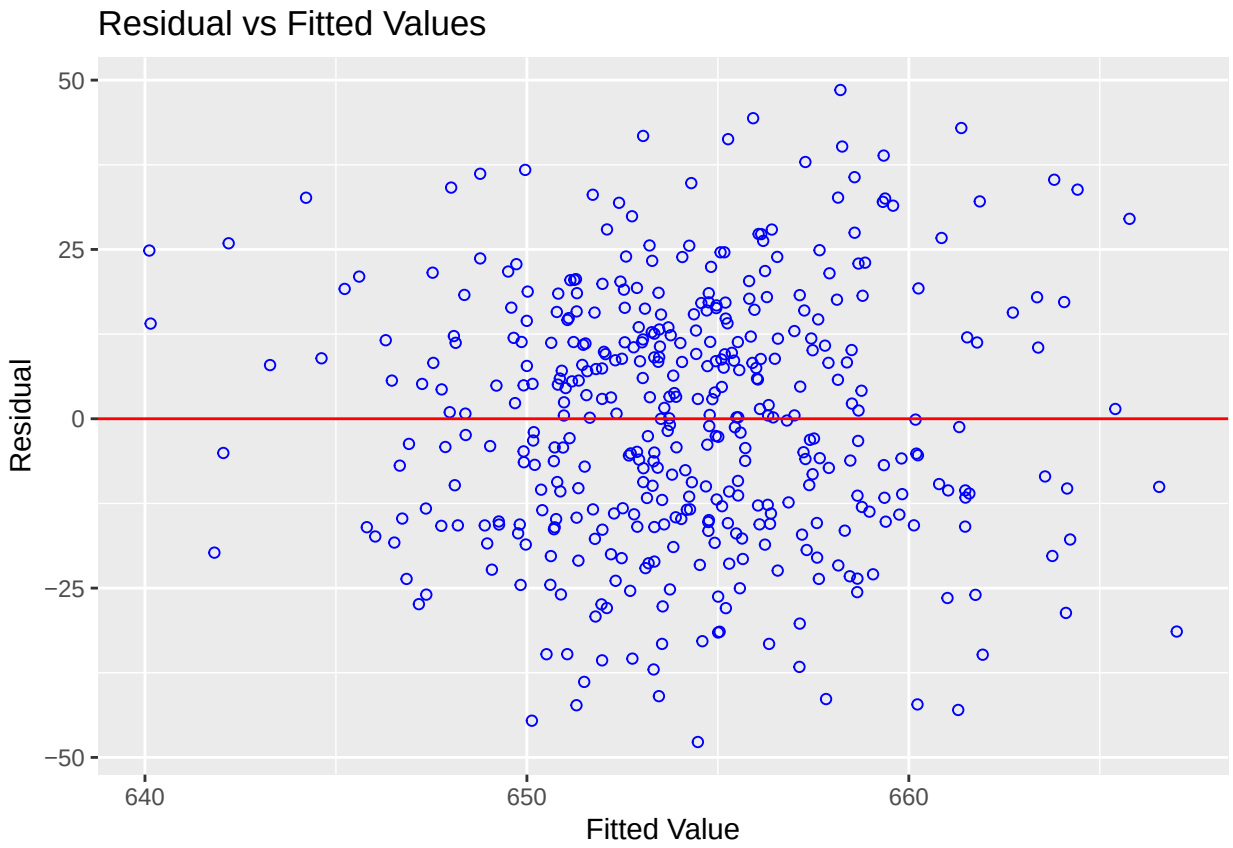

```
ols_test_normality(reg_linear)
```

```
## -----------------------------------------------
##        Test             Statistic       pvalue
## -----------------------------------------------
## Shapiro-Wilk              0.9944         0.1249
## Kolmogorov-Smirnov        0.045          0.3632
## Cramer-von Mises          32.948         0.0000
## Anderson-Darling          0.7869         0.0410
## -----------------------------------------------
```

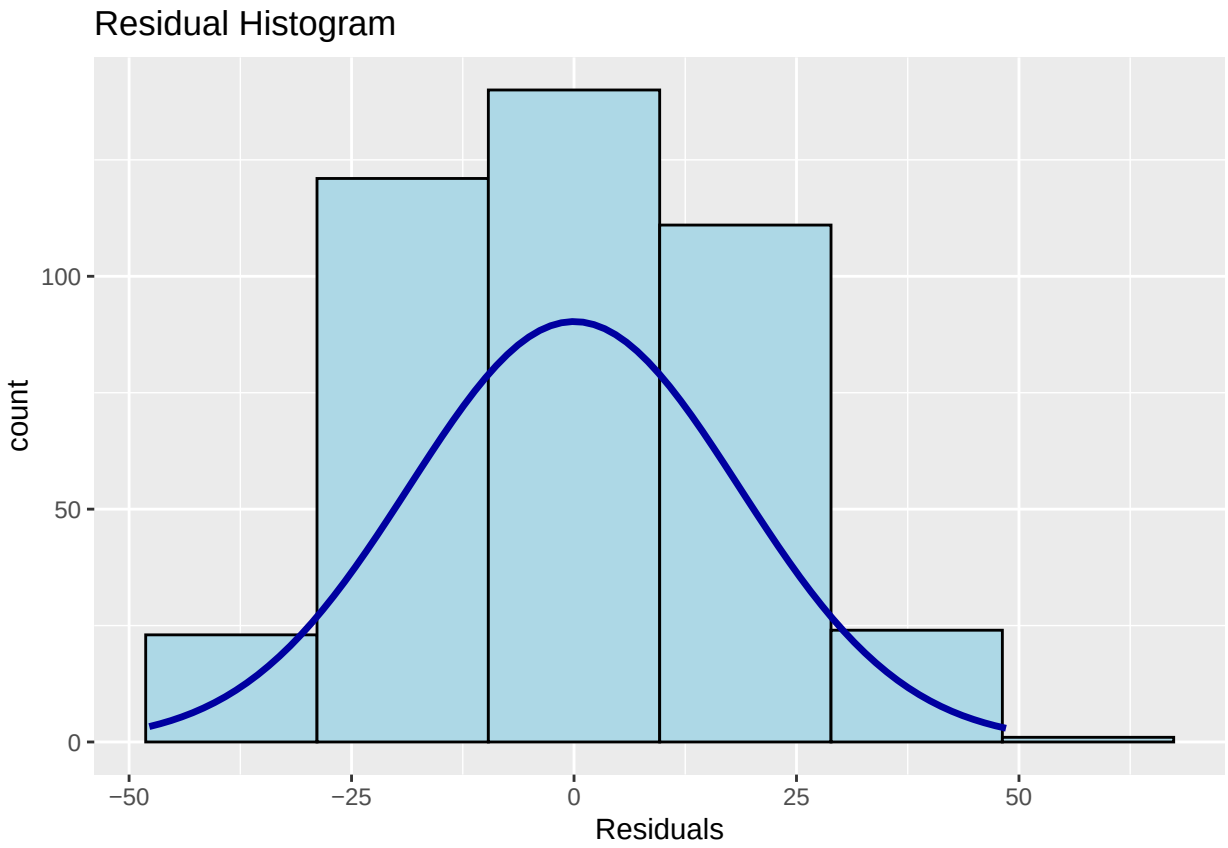
```
ols_test_correlation(reg_linear)
```

```
## [1] 0.997736
```

```
ols_plot_resid_fit(reg_linear)
```




```
ols_plot_resid_hist(reg_linear)
```



Testes de Heterocedasticidade

```
lmtest::bptest(reg_linear)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  reg_linear
## BP = 5.7936, df = 1, p-value = 0.01608
```

```
car::ncvTest(reg_linear)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 4.483477, Df = 1, p = 0.034224
```

Corrigindo as Estimações para Heterocedasticidade

```
coeftest(reg_linear, vcov = vcovHC(reg_linear, "HC1"))
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 698.93295   10.36436 67.4362 < 2.2e-16 ***
## STR          -2.27981    0.51949 -4.3886 1.447e-05 ***
## ---
```

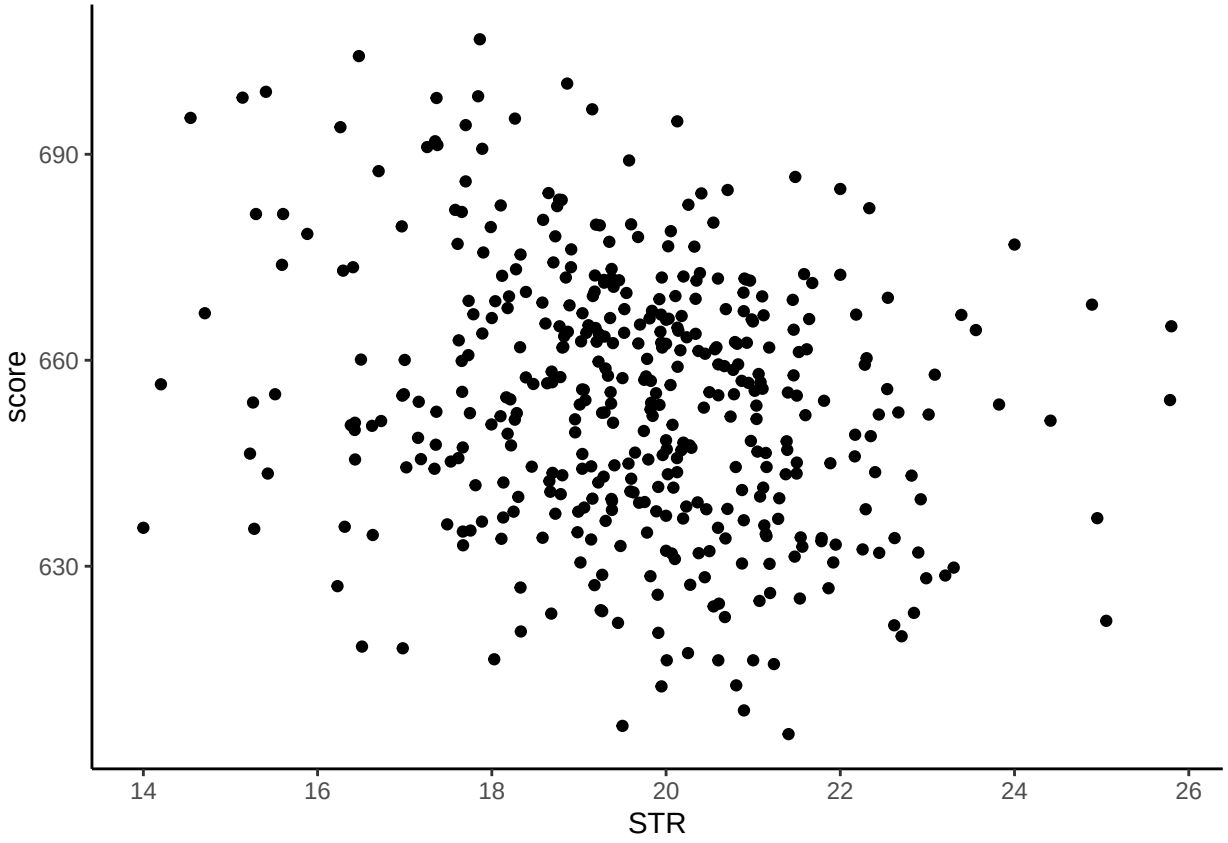
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plotando as observações em um Gráfico

Agora iremos plotar os dados e o modelo estimado em um gráfico.

1º plotamos o gráfico só com as observações:

```
data.graph = ggplot(CASchools, aes(x=STR, y=score))+
                     geom_point() +
                     theme_classic()
data.graph
```

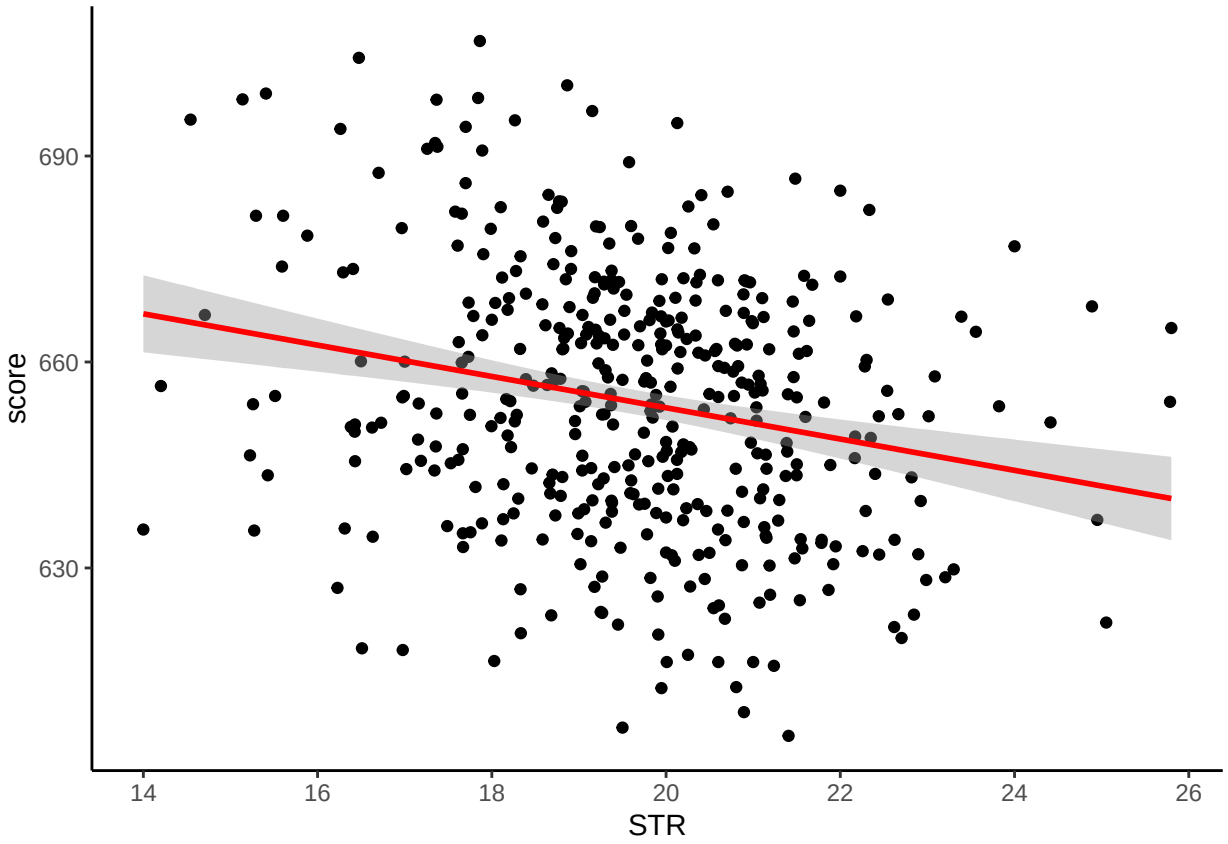


Agora plotamos o gráfico com uma tendência linear, que é exatamente o que a regressão faz:

```
data.graph = data.graph +
  geom_smooth(method="lm", col="red", level=0.95)

data.graph
```

```
## `geom_smooth()` using formula 'y ~ x'
```

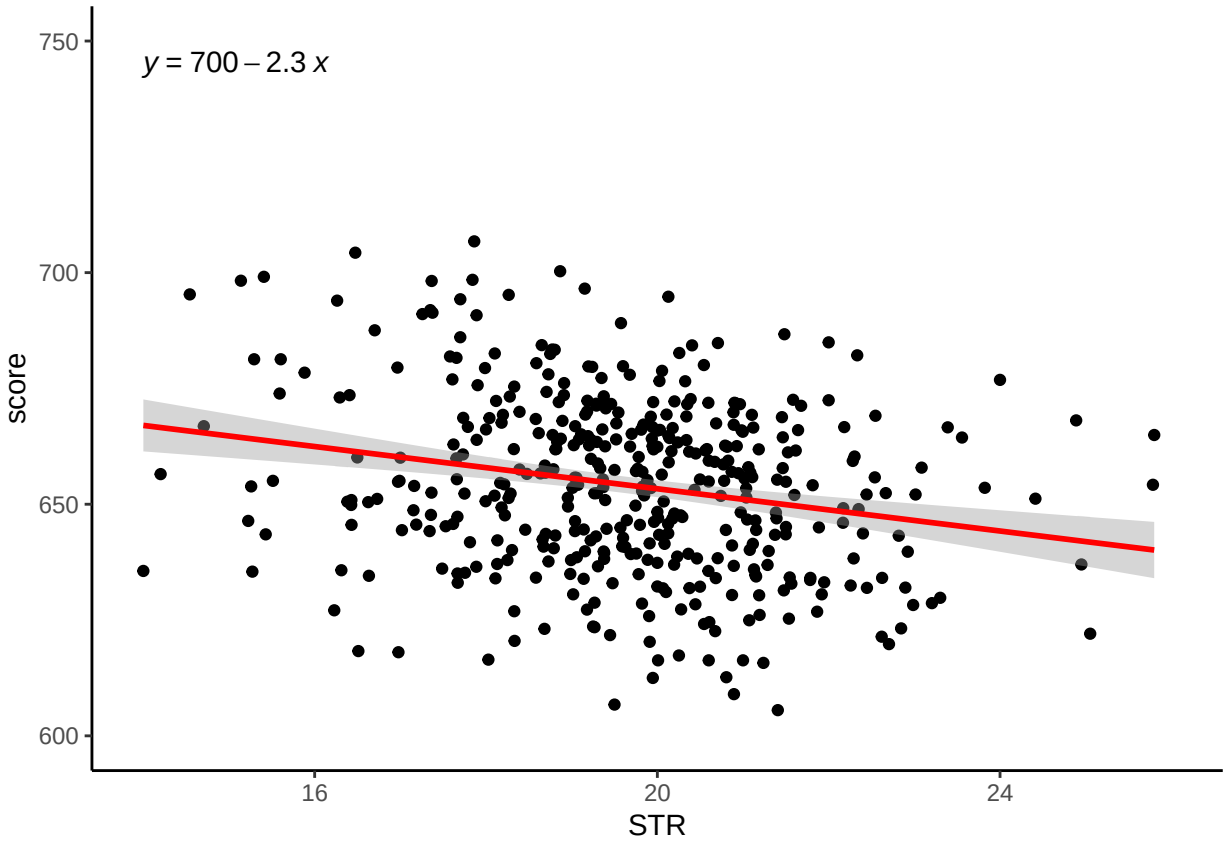


Por fim plotamos o gráfico com a regressão linear proposta

```
data.graph <- data.graph +
  stat_regline_equation() +
  xlim(14, 26) +
  ylim(600, 750)

data.graph
```

```
## `geom_smooth()` using formula 'y ~ x'
```


y = 700 – 2.3 x
750
700
650
600
score
16
20
24
STR