Aula 14 - Regressão e Correlação no R

Prof. Weligton Gomes

2023-06-08

Regressão Linear Simples:

Análise de regressão é uma técnica estatística utilizada para investigar a relação existente entre variáveis através da construção de uma equação (um modelo). De maneira geral, essa técnica pode ser utilizada com vários objetivos, dentre os quais se pode destacar: descrever a relação entre variáveis para entender um processo ou fenômeno; prever o valor de uma variável a partir do conhecimento dos valores das outras variáveis; substituir a medição de uma variável pela observação dos valores de outras variáveis; controlar os valores de uma variável em uma faixa de interesse.

Estudo Empírico do pacote ISwR

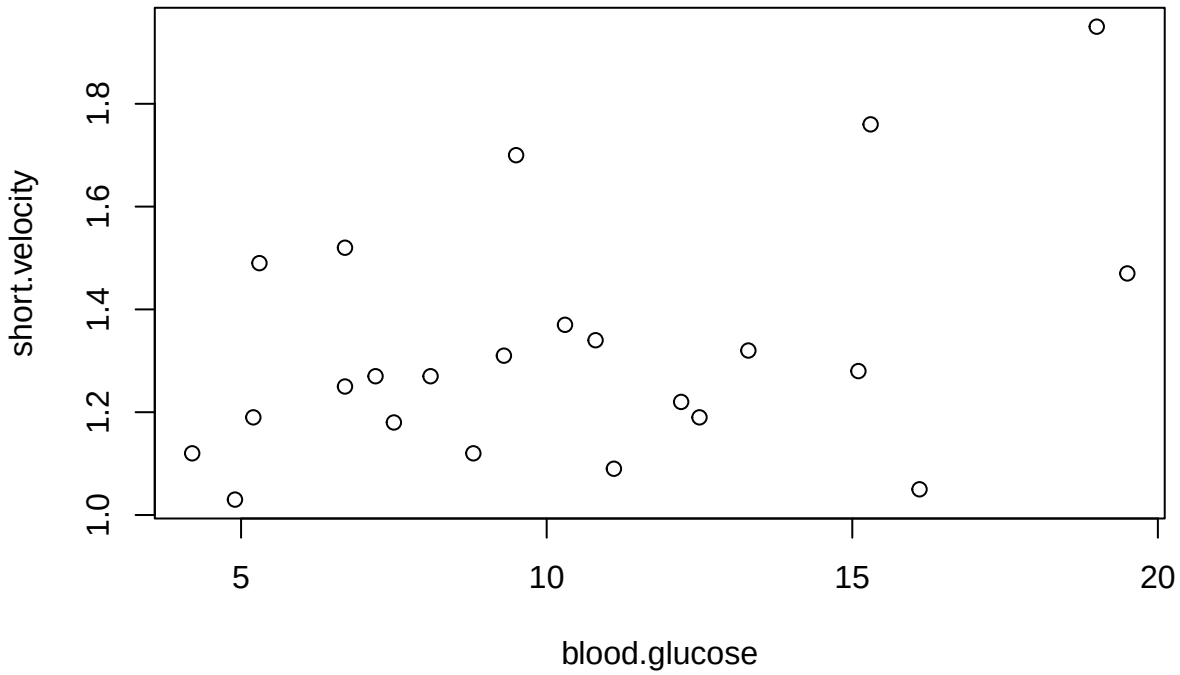
Pergunta: Existe uma relação entre a glicemia de jejum e a velocidade de encurtamento ventricular em pacientes diabéticos tipo 1? Em caso afirmativo, qual é a natureza da associação?

De outra forma, você pode estar interessado em descrever a velocidade de encurtamento circunferencial média (%/s) (short.velocity), ou seja, a velocidade na qual um músculo muda de comprimento durante uma contração, como uma função da glicose no sangue em jejum ($mmol/l$) (blood.glucose).

Vejamos as estatísticas descritivas das variáveis.

```
library(ISwR)
attach(thuesen)
summary(thuesen)
```

```
##  blood.glucose    short.velocity
##  Min.   : 4.200   Min.   :1.030
##  1st Qu.: 7.075   1st Qu.:1.185
##  Median : 9.400   Median :1.270
##  Mean   :10.300   Mean   :1.326
##  3rd Qu.:12.700   3rd Qu.:1.420
##  Max.   :19.500   Max.   :1.950
##                   NA's   :1
```



Descrição da Função lm (linear model)

Para análise de regressão linear, utiliza-se a função lm (modelo linear). O argumento principal desta função é a fórmula na qual a variável dependente é seguida pelo símbolo do til e o somatório das variáveis explicativas.

```
attach(thuesen)
```

```
## The following objects are masked from thuesen (pos = 3):
##
##     blood.glucose, short.velocity
```

```
lm(short.velocity~blood.glucose)
```

```
##
## Call:
## lm(formula = short.velocity ~ blood.glucose)
##
## Coefficients:
##   (Intercept)  blood.glucose
##       1.09781        0.02196
```

```
lm.velo<-lm(short.velocity~blood.glucose)
summary(lm.velo)
```

```
##
## Call:
## lm(formula = short.velocity ~ blood.glucose)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40141 -0.14760 -0.02202  0.03001  0.43490
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)    1.09781    0.11748   9.345 6.26e-09 ***
## blood.glucose  0.02196    0.01045   2.101   0.0479 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2167 on 21 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.1737, Adjusted R-squared:  0.1343
## F-statistic: 4.414 on 1 and 21 DF,  p-value: 0.0479
```

Resíduos e Valores Ajustados

Após a estimação, obtém-se os valores ajustados e os resíduos a partir das funções fitted e resid.

Ao conduzir uma análise residual, um "gráfico de resíduos versus ajustes" é o gráfico criado com mais frequência. É um gráfico de dispersão de resíduos no eixo y e valores ajustados (respostas estimadas) no eixo x. O gráfico é usado para detectar não linearidade, variâncias de erro desiguais e outliers.

Valores Ajustados (Valor previsto de y):

```
fitted(lm.velo)
```

```
##        1        2        3        4        5        6        7        8
## 1.433841 1.335010 1.275711 1.526084 1.255945 1.214216 1.302066 1.341599
##        9       10       11       12       13       14       15       17
## 1.262534 1.365758 1.244964 1.212020 1.515103 1.429449 1.244964 1.190057
##       18       19       20       21       22       23       24
## 1.324029 1.372346 1.451411 1.389916 1.205431 1.291085 1.306459
```

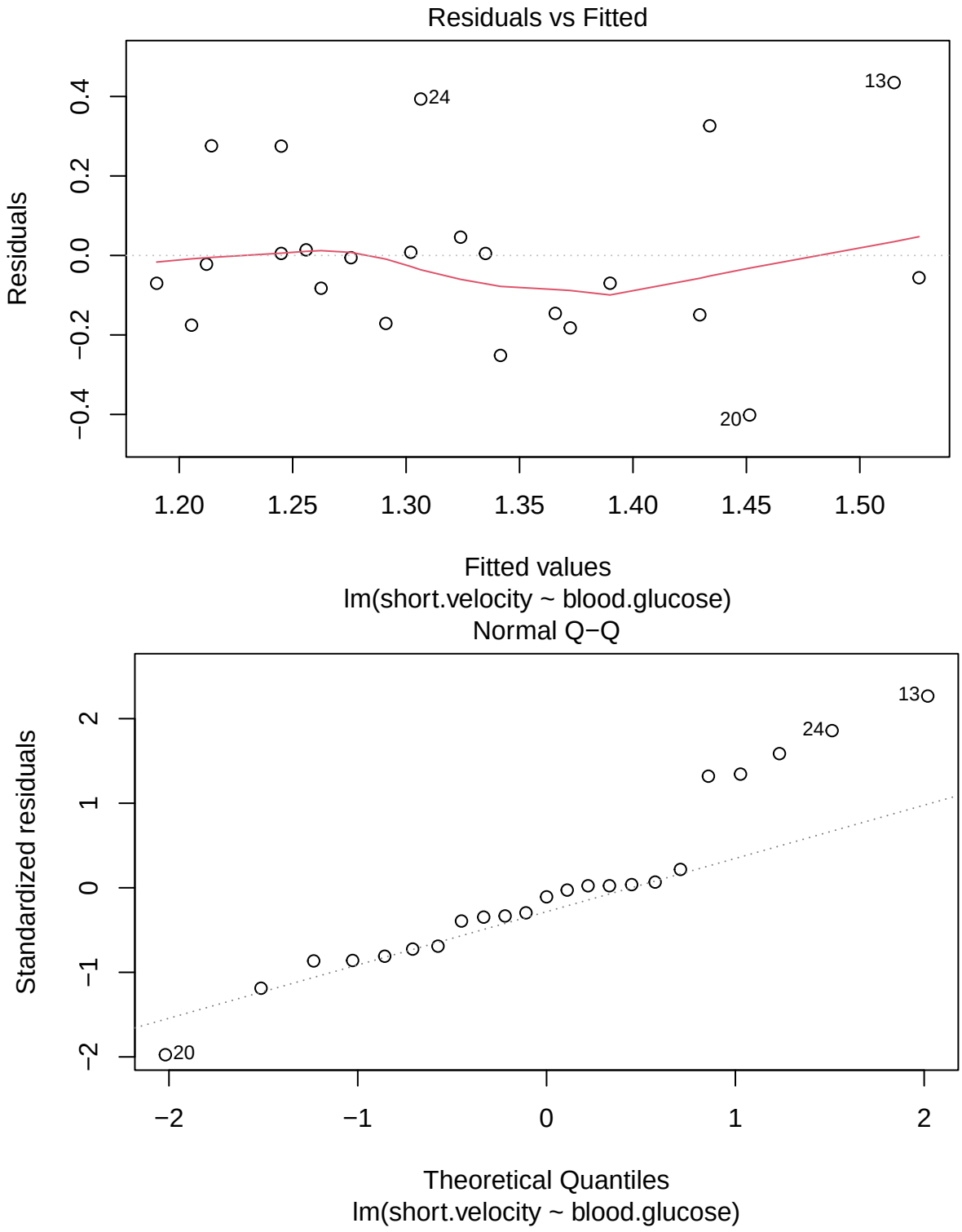
Resíduos

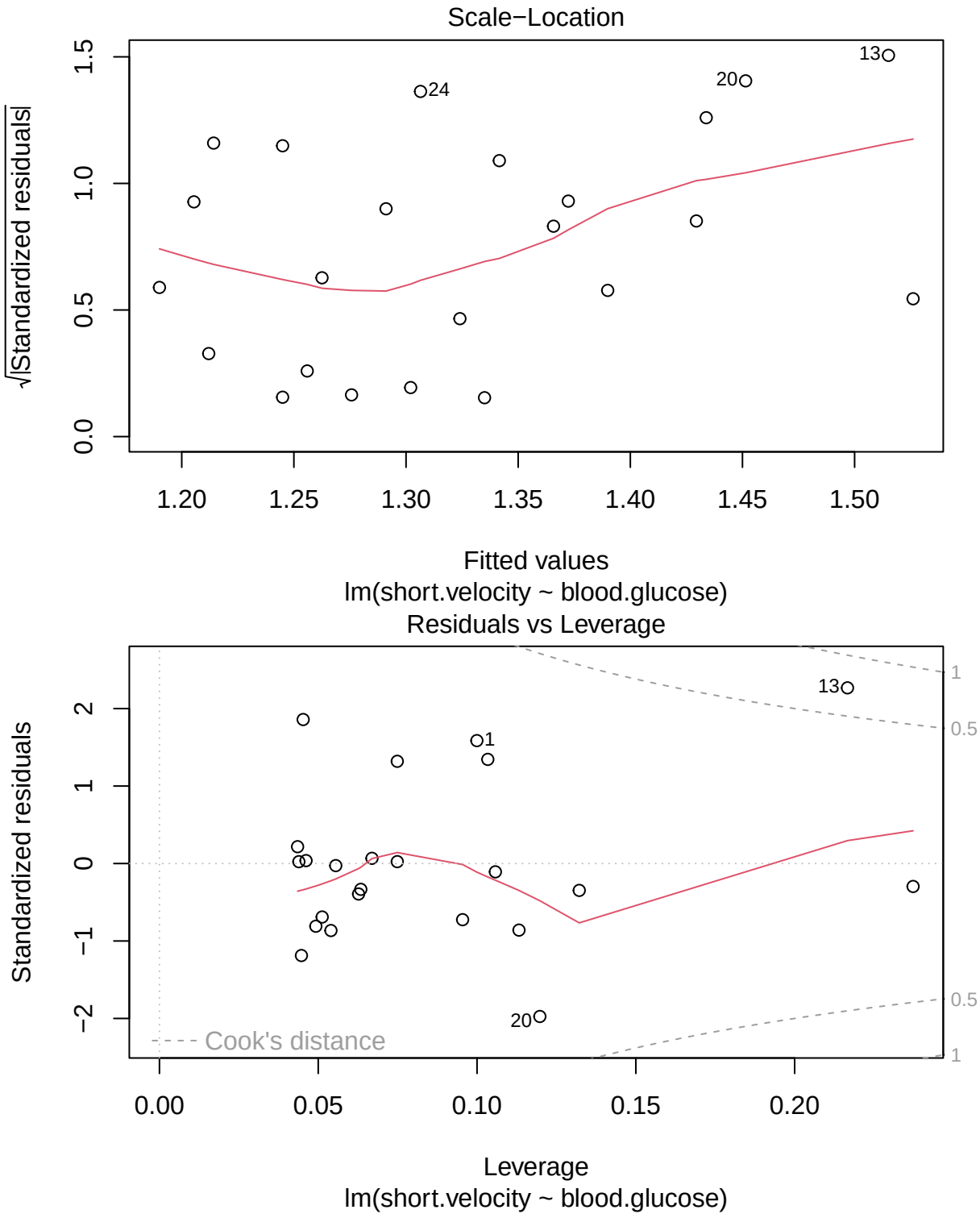
```
resid(lm.velo)
```

```
##            1            2            3            4            5            6
##  0.326158532  0.004989882 -0.005711308 -0.056084062  0.014054962  0.275783754
##            7            8            9           10           11           12
##  0.007933665 -0.251598875 -0.082533795 -0.145757649  0.005036223 -0.022019994
##           13           14           15           17           18           19
##  0.434897199 -0.149448964  0.275036223 -0.070057471  0.045971143 -0.182346406
##           20           21           22           23           24
## -0.401411486 -0.069916424 -0.175431237 -0.171085074  0.393541161
```

Plotando os gráficos da regressão.

```
plot(lm.velo)
```


Residuals vs Fitted
Residuals
Fitted values
lm(short.velocity ~ blood.glucose)
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(short.velocity ~ blood.glucose)
13
24
20



Observações:

1) O primeiro gráfico (resíduos vs. valores ajustados) é um gráfico de dispersão simples entre resíduos e valores previstos. Deve parecer mais ou menos aleatório.

2) O segundo gráfico (Q-Q normal) é um gráfico de probabilidade normal. Ele dará uma linha reta se os erros forem distribuídos normalmente, mas os pontos 13, 20 e 24 desviam da linha reta.

3) O terceiro gráfico (Escala-Localização), como o primeiro, deve parecer aleatório. Sem padrões. O nosso temos um estranho padrão em forma de V. Também é chamado de gráfico Spread-Location. Este gráfico mostra se os resíduos são distribuídos igualmente ao longo dos intervalos de preditores. É assim que você pode verificar a suposição de variância igual (homocedasticidade). É bom se você vir uma linha horizontal com pontos de propagação igualmente (aleatoriamente).

4) O último gráfico (distância de Cook) nos diz quais pontos têm a maior influência na regressão (pontos de alavancagem). Vemos que os pontos 1, 13 e 20 têm grande influência no modelo.

Este gráfico pode ser usado para encontrar casos influentes no conjunto de dados. Um caso influente é aquele que, se removido, afetará o modelo, portanto, sua inclusão ou exclusão deve ser considerada.

Um caso influente pode ou não ser um outlier e o objetivo deste gráfico é identificar os casos que têm alta influência no modelo. Os valores discrepantes tendem a exercer influência e, portanto, influenciar o modelo.

Quando os pontos estão fora da distância do Cook, isso significa que eles têm altas pontuações de distância do Cook. Nesse caso, os valores influenciam os resultados da regressão. Os resultados da regressão serão alterados se excluirmos esses casos.

Correlação

Um coeficiente de correlação é uma medida simétrica e invariante de escala de associação entre duas variáveis aleatórias. Ele varia de -1 a $+1$, onde os extremos indicam correlação perfeita e 0 significa nenhuma correlação. O sinal é negativo quando grandes valores de uma variável estão associados a pequenos valores da outra e positivo se ambas as variáveis tendem a ser grandes ou pequenas simultaneamente.

Todas as funções estatísticas elementares em R requerem que todos os valores sejam não omissos ou que você declare explicitamente o que deve ser feito com os casos com valores omissos. Para médias, var, sd e funções similares de um vetor, você pode fornecer o argumento $na.rm = T$ para indicar que os valores ausentes devem ser removidos antes do cálculo.

```
attach(thuesen)
```

```
## The following objects are masked from thuesen (pos = 3):
##
##     blood.glucose, short.velocity
```

```
## The following objects are masked from thuesen (pos = 4):
##
##     blood.glucose, short.velocity
```

```
cor(blood.glucose,short.velocity)
```

```
## [1] NA
```

```
cor(blood.glucose,short.velocity,use="complete.obs")
```

```
## [1] 0.4167546
```

```
cor(thuesen,use="complete.obs")
```

```
##                blood.glucose short.velocity
## blood.glucose      1.0000000      0.4167546
## short.velocity     0.4167546      1.0000000
```

No entanto, os cálculos acima não fornecem nenhuma indicação se a correlação é significativamente diferente de zero.

Para isso, você precisa de cor.test. Funciona simplesmente especificando as duas variáveis:

```
cor.test(blood.glucose,short.velocity)
```

```
##
##  Pearson's product-moment correlation
##
## data:  blood.glucose and short.velocity
## t = 2.101, df = 21, p-value = 0.0479
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.005496682 0.707429479
## sample estimates:
##       cor
## 0.4167546
```

Correlação de Spearman's ρ

a correlação de Spearman descreve a relação entre as variáveis através de uma função monotética, ou seja, uma classificação que utiliza apenas um critério diferenciador, por exemplo, as caracetristicas partilhadas por uma população de uma mesma cultura.

Isso significa, de maneira simplificada, que ele está analisando se, quando o valor de uma variável aumenta ou diminui, o valor da outra variável aumenta ou diminui.

Muitas vezes podemos estar interessado em variantes não paramétricas. Estas têm a vantagem de não depender da distribuição normal e, de fato, serem invariantes às transformações monótonas das coordenadas.

A principal desvantagem é que sua interpretação não é muito clara. Uma escolha popular e simples é o coeficiente de correlação de posto de Spearman ρ. Isso é obtido simplesmente substituindo as observações por sua classificação e computando a correlação. Sob a hipótese nula de independência entre as duas variáveis, a distribuição exata de ρ pode ser calculada.

```
cor.test(blood.glucose,short.velocity,method="spearman")
```

```
## Warning in cor.test.default(blood.glucose, short.velocity, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  blood.glucose and short.velocity
## S = 1380.4, p-value = 0.1392
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.318002
```

Correlação de Kendall's τ

O teste τ de Kendall se baseia na contagem do número de pares concordantes e discordantes. Um par de pontos é concordante se a diferença na coordenada x for do mesmo sinal que a diferença na coordenada y. Para uma relação monótona perfeita, ou todos os pares serão concordantes ou todos os pares serão discordantes.

Sob a independência, deve haver tantos pares concordantes quanto discordantes.

```
cor.test(blood.glucose,short.velocity,method="kendall")
```

```
## Warning in cor.test.default(blood.glucose, short.velocity, method = "kendall"):
## Cannot compute exact p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  blood.glucose and short.velocity
## z = 1.5604, p-value = 0.1187
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.2350616
```

```
detach(thuesen)
library(wooldridge)
```

Base de Dados do Wooldridge

1 - O data.frame contém 156 observações de 21 variáveis. 2 - O dataset está no pacote wooldridge acessado fazendo

```
data("lawsch85")
View(lawsch85)
```

Variável Dependente:

1) salary: median starting salary

Características dos Entrantes

2) LSAT: median LSAT (Law School Admission Test) score
3) GPA: median college GPA (Grade Point Average)

Características da Escola

4) rank: law school ranking
5) cost: law school cost
6) libvol: no. volumes in lib., 1000s

Variáveis Derivadas

7) lsalary: log(salary)
8) top10: = 1 if ranked in top 10
9) r11_25: = 1 if ranked 11-25
10) r26_40: = 1 if ranked 26-40
11) r41_60: = 1 if ranked 41-60
12) llibvol: log(libvol)
13) lcost: log(cost)

```
attach(lawsch85)

#Criando variável Dummy para o rank entre 61-100
r61.100 <- as.numeric(lawsch85$rank >= 61 & lawsch85$rank <= 100)
lawsch85<-as.data.frame(cbind(lawsch85,r61.100))
```

```r
mod1 <- lm(log(salary) ~ log(rank) + log(LSAT) + log(GPA) + log(libvol) + log(cost),
    data = lawsch85)
summary(mod1)
```

```
##
## Call:
## lm(formula = log(salary) ~ log(rank) + log(LSAT) + log(GPA) +
##     log(libvol) + log(cost), data = lawsch85)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43582 -0.05669 -0.00996  0.03835  0.29717
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.382899   2.576873   2.865  0.00486 **
## log(rank)   -0.226917   0.018967 -11.964  < 2e-16 ***
## log(LSAT)    0.725794   0.558971   1.298  0.19643
## log(GPA)     0.191409   0.281913   0.679  0.49837
## log(libvol)  0.016899   0.032585   0.519  0.60492
## log(cost)    0.008086   0.029495   0.274  0.78442
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1017 on 130 degrees of freedom
##   (20 observations deleted due to missingness)
## Multiple R-squared:  0.8704, Adjusted R-squared:  0.8654
## F-statistic: 174.6 on 5 and 130 DF,  p-value: < 2.2e-16
```

```r
mod2 <- lm((salary) ~ (rank) + (LSAT) + (GPA) + (libvol) + (cost), data = lawsch85)
mod3 <- lm(log(salary) ~ top10 + r11_25 + r26_40 + r41_60 + r61.100 + log(LSAT) +
    log(GPA) + log(libvol) + log(cost), data = lawsch85)

#AIC - Teste de Akaike e BIC - Teste de Schwarz
mod1$AIC <- AIC(mod1)
mod2$AIC <- AIC(mod2)
mod3$AIC <- AIC(mod3)
mod1$BIC <- BIC(mod1)
mod2$BIC <- BIC(mod2)
mod3$BIC <- BIC(mod3)

library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```r
star.1 <- stargazer(mod1, mod2, mod3, title = "Título: Resultados das Regressões",
    column.labels = c("MQO-mod1", "MQO-mod2", "MQO-mod3"), align = TRUE, type = "text",
    keep.stat = c("aic", "bic", "rsq", "adj.rsq", "n"))
```

```
##
## Título: Resultados das Regressões
```

```
## ===========================================================
##                              Dependent variable:
##                     ------------------------------------
##                     log(salary)   (salary)    log(salary)
##                      MQO-mod1     MQO-mod2     MQO-mod3
##                         (1)          (2)          (3)
## -----------------------------------------------------------
## log(rank)            -0.227***
##                       (0.019)
##
## top10                                           0.700***
##                                                  (0.053)
##
## r11_25                                          0.593***
##                                                  (0.039)
##
## r26_40                                          0.374***
##                                                  (0.034)
##
## r41_60                                          0.262***
##                                                  (0.028)
##
## r61.100                                         0.131***
##                                                  (0.021)
##
## log(LSAT)              0.726                     0.885*
##                       (0.559)                   (0.472)
##
## log(GPA)               0.191                     0.057
##                       (0.282)                   (0.239)
##
## log(libvol)            0.017                     0.036
##                       (0.033)                   (0.026)
##
## log(cost)              0.008                     0.001
##                       (0.029)                   (0.025)
##
## rank                              -115.815***
##                                    (16.153)
##
## LSAT                                155.286
##                                    (192.598)
##
## GPA                               14,500.240***
##                                    (4,357.070)
##
## libvol                              12.956***
##                                     (3.261)
##
## cost                                 0.366**
##                                     (0.146)
##
## Constant              7.383***     -33,124.550    5.562**
##                        (2.577)    (25,210.760)    (2.141)
```

```
##
## ---------------------------------------------------------
## Observations            136          136          136
## R2                    0.870        0.806        0.911
## Adjusted R2           0.865        0.798        0.905
## Akaike Inf. Crit.  -227.844    2,737.692    -271.086
## Bayesian Inf. Crit. -207.455    2,758.080    -239.046
## =========================================================
## Note:                         *p<0.1; **p<0.05; ***p<0.01
```

Teste de Multicolinearidade (vif)

```
library(car)
```

```
## Loading required package: carData
```

```
reg1.vif <- vif(mod1)
reg1.vif
```

```
##   log(rank)   log(LSAT)    log(GPA) log(libvol)   log(cost)
##    4.676860    3.456882    3.636916    2.475024    1.621828
```

```
reg2.vif <- vif(mod2)
reg2.vif
```

```
##     rank     LSAT      GPA   libvol     cost
## 2.777994 3.469461 3.264910 1.735231 1.558141
```

```
reg3.vif <- vif(mod3)
reg3.vif
```

```
##       top10      r11_25      r26_40      r41_60     r61.100   log(LSAT)
##    3.488650    2.635882    1.730471    1.589780    1.542470    3.478267
##    log(GPA) log(libvol)   log(cost)
##    3.693818    2.225695    1.653360
```

Heterocedasticidade no modelo 3

Teste de White

```
lmtest::bptest(mod3, ~log(LSAT) + log(GPA) + log(libvol) + log(cost) + I(log(LSAT)^2) +
    I(log(GPA)^2) + I(log(libvol)^2) + I(log(cost)^2), data = lawsch85)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mod3
## BP = 15.673, df = 8, p-value = 0.04731
```

Podemos concluir que há a presença de heterocedasticidade dos resíduos em mod3.

Autocorrelação dos resíduos

```
library(car)
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

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

```
library(sandwich)

dw.mod3 <- dwtest(mod3)
dw.mod3
```

```
##
##  Durbin-Watson test
##
## data:  mod3
## DW = 1.9225, p-value = 0.3026
## alternative hypothesis: true autocorrelation is greater than 0
```

Teste de Jarque-Bera para normalidade

```
u.hat <- resid(mod3)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
JB.mod3 <- jarque.bera.test(u.hat)
JB.mod3
```

```
##
##  Jarque Bera Test
##
## data:  u.hat
## X-squared = 20.538, df = 2, p-value = 3.47e-05
```

Rejeita-se H0 e conclui-se por resíduos não normais.