Data Science - Aprofundamento: trilha 4

Rafael Garcia de Lima

2021

Importação das bibliotecas

```
library(DT);
library(gvlma);
library(dplyr);
library(ggpubr);
library(ggplot2);
library(dataMeta);
library(tidyverse);
```

Problema 1

Iniciando os dados

```
y <- c(23, 21, 20, 19, 17, 16, 16, 15, 13);
x <- c(3.5, 3.7, 3.8, 4.2, 4.6, 4.7, 4.9, 5.2, 5.5);
```

Criando data frame

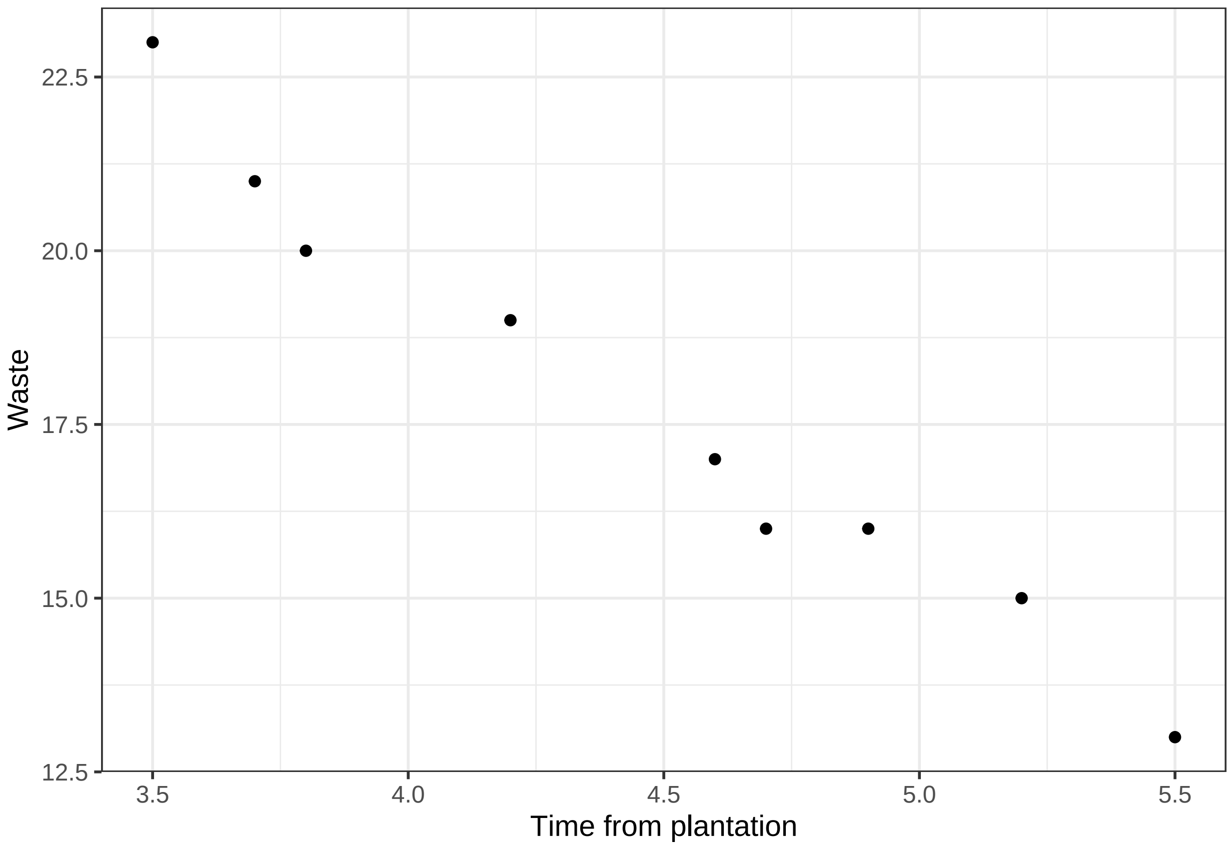
```
df <- data.frame(tempplant = x, waste=y);
df;
```

```
##   tempplant waste
## 1       3.5    23
## 2       3.7    21
## 3       3.8    20
## 4       4.2    19
## 5       4.6    17
## 6       4.7    16
## 7       4.9    16
## 8       5.2    15
## 9       5.5    13
```

a. Gráfico de dispersão

Gráfico de disperção entre o tempo de plantio e a perda

```r
disp <- ggplot(df, aes(x=tempplant, y=waste)) + geom_point() + labs(x="Time from plantation", y="Waste"
disp;
```



b. Modelo de regressão linear

```r
fit <- lm(waste ~ tempplant, data = df);
fit;
```

```
##
## Call:
## lm(formula = waste ~ tempplant, data = df)
##
## Coefficients:
## (Intercept)    tempplant
##      37.861       -4.507
```

c. análise do modelo

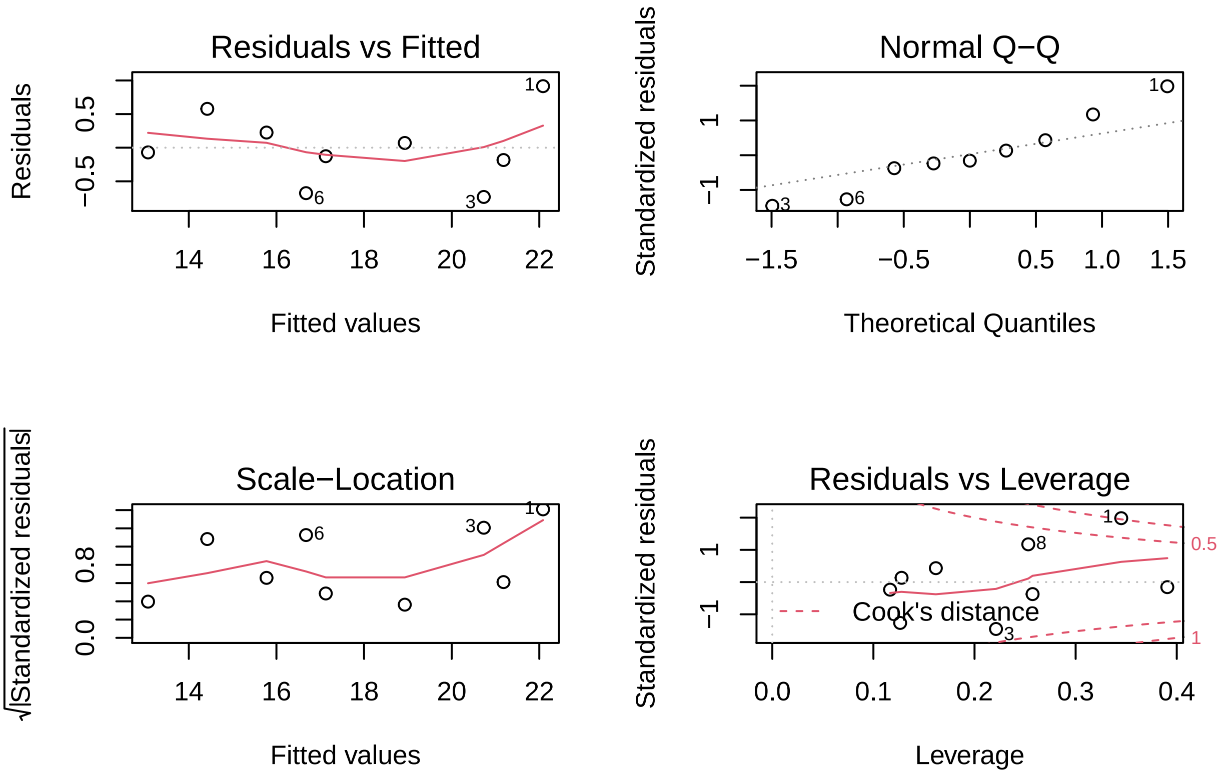
c.i Sumário do modelo

```r
summary(fit);
```

```
##
## Call:
## lm(formula = waste ~ tempplant, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73263 -0.18337 -0.07005  0.22551  0.91515
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.8608     1.2999   29.13 1.45e-08 ***
## tempplant    -4.5074     0.2886  -15.62 1.07e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5701 on 7 degrees of freedom
## Multiple R-squared:  0.9721, Adjusted R-squared:  0.9681
## F-statistic: 243.9 on 1 and 7 DF,  p-value: 1.067e-06
```

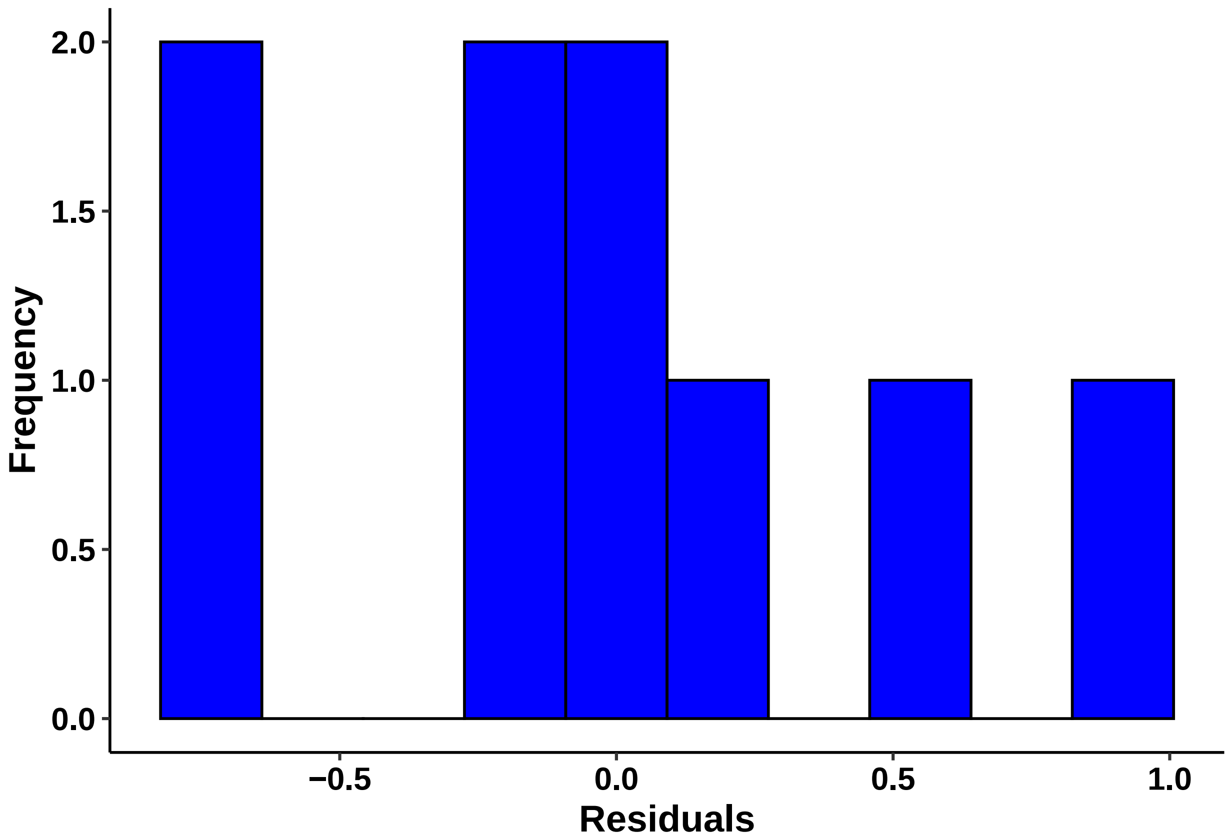
c.ii Gráficos de diagnóstico

```r
par(mfrow = c(2, 2));
plot(fit);
```



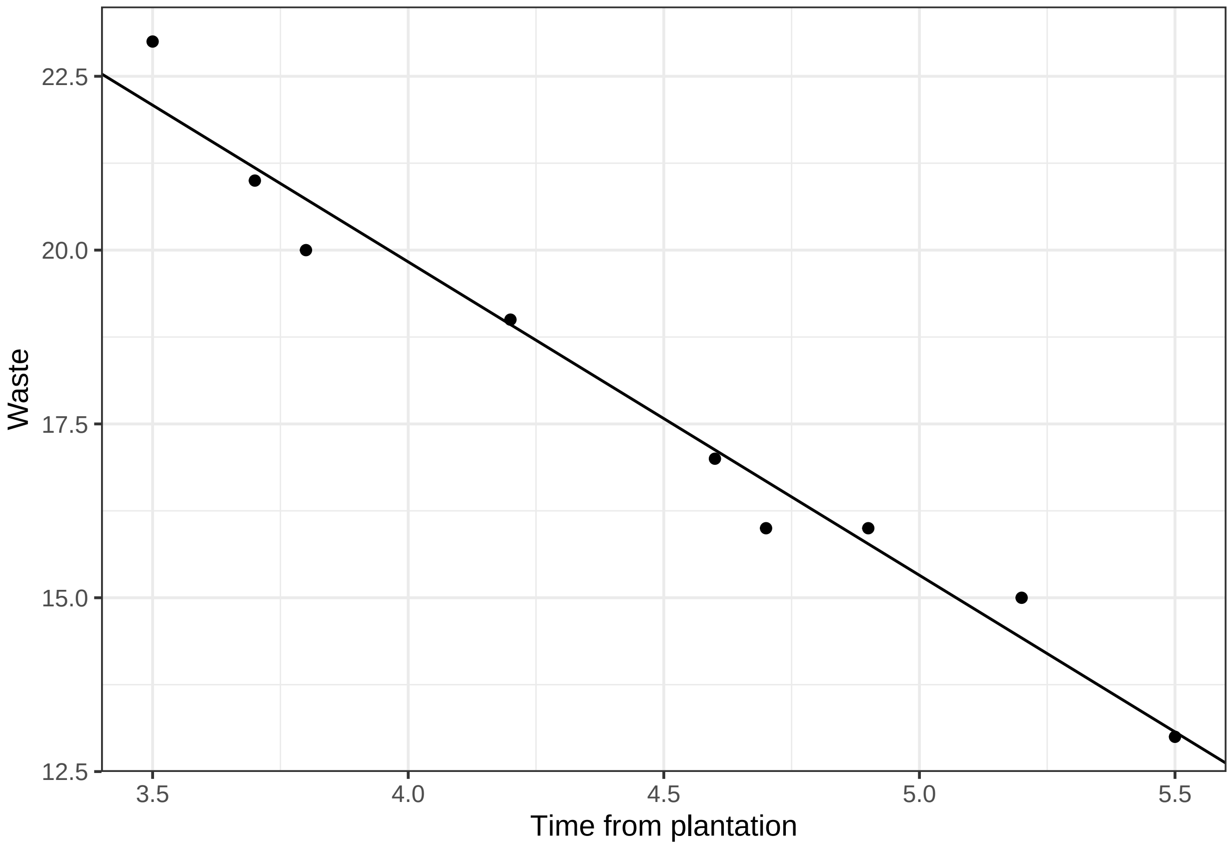
Topo-esquerda Sem estruturas ou padrões, o ajuste da reta é necessário #### Topo-direita Os erros são aleatóriamente distrubuídos

```
fit.df <- as.data.frame(residuals(fit));
names(fit.df) <- c("residuals");
aResiduals <- ggplot(fit.df, aes(residuals)) +
  geom_histogram(fill="blue", color="black", stat = "bin", bins = 10) +
  theme_pubr() + labs_pubr() + xlab("Residuals") + ylab("Frequency") +
  theme(plot.caption = element_text(hjust = 0));
aResiduals;
```



d. Adição da reta de regressão

```
cofs <- coef(fit);
disp <- disp + geom_abline(aes(intercept=cofs[1], slope=cofs[2]));
disp
```



Problema 2

a. Importação e sumário

```
pib <- read.csv("pib_gapminder.csv");
summary(pib);
```

```
##      pais                ano            pop              continente
##  Length:1704        Min.   :1952   Min.   :6.001e+04   Length:1704
##  Class :character   1st Qu.:1966   1st Qu.:2.794e+06   Class :character
##  Mode  :character   Median :1980   Median :7.024e+06   Mode  :character
##                     Mean   :1980   Mean   :2.960e+07
##                     3rd Qu.:1993   3rd Qu.:1.959e+07
##                     Max.   :2007   Max.   :1.319e+09
##     expVida         pibPercap
##  Min.   :23.60   Min.   :   241.2
##  1st Qu.:48.20   1st Qu.:  1202.1
##  Median :60.71   Median :  3531.8
##  Mean   :59.47   Mean   :  7215.3
##  3rd Qu.:70.85   3rd Qu.:  9325.5
##  Max.   :82.60   Max.   :113523.1
```

b. Estrutura dos dados

```
str(pib);
```

```
## 'data.frame':    1704 obs. of  6 variables:
##  $ pais      : chr  "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
##  $ ano       : int  1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 ...
##  $ pop       : num  8425333 9240934 10267083 11537966 13079460 ...
##  $ continente: chr  "Asia" "Asia" "Asia" "Asia" ...
##  $ expVida   : num  28.8 30.3 32 34 36.1 ...
##  $ pibPercap : num  779 821 853 836 740 ...
```

c. Classificação das variáveis

pais: qualitativa nominal

ano: quantitativa contínua

pop: quantitativa discreta

continente: qualitativa nominal

expVida: quantitativa discreta

pibPercap: quantitativa contínua

d. Sumário dos dados

```
summary(pib);
```

```
##      pais                ano            pop               continente
##  Length:1704        Min.   :1952   Min.   :6.001e+04   Length:1704
##  Class :character   1st Qu.:1966   1st Qu.:2.794e+06   Class :character
##  Mode  :character   Median :1980   Median :7.024e+06   Mode  :character
##                     Mean   :1980   Mean   :2.960e+07
##                     3rd Qu.:1993   3rd Qu.:1.959e+07
##                     Max.   :2007   Max.   :1.319e+09
##     expVida        pibPercap
##  Min.   :23.60   Min.   :   241.2
##  1st Qu.:48.20   1st Qu.:  1202.1
##  Median :60.71   Median :  3531.8
##  Mean   :59.47   Mean   :  7215.3
##  3rd Qu.:70.85   3rd Qu.:  9325.5
##  Max.   :82.60   Max.   :113523.1
```

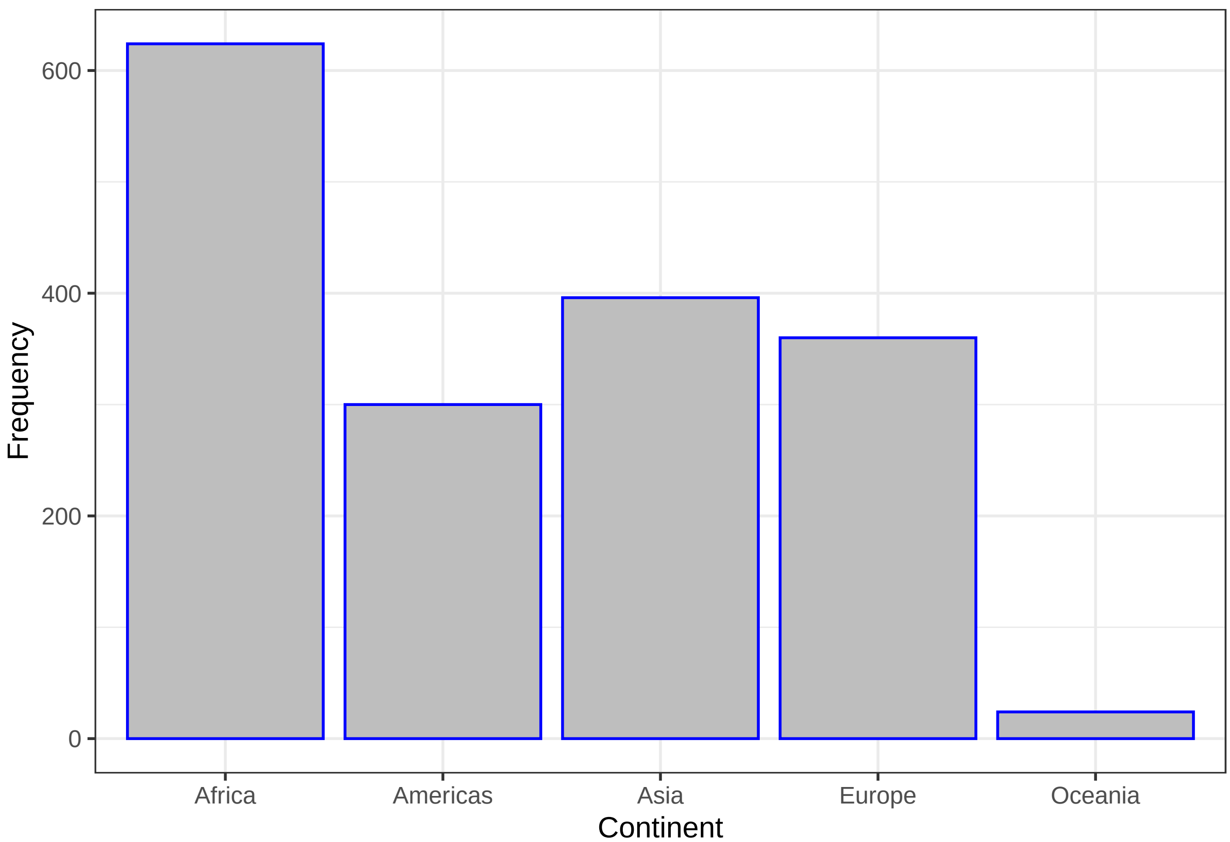
e. Tabelas de frequência absoluta e frequência relativa

```
data <- pib %>% select(continente) %>%
  group_by(continente) %>%
  summarise(absF=n()) %>%
  mutate(relF = 100 * absF / sum(absF));
data;
```

```
## # A tibble: 5 x 3
##   continente  absF  relF
##   <chr>      <int> <dbl>
## 1 Africa       624 36.6
## 2 Americas     300 17.6
## 3 Asia         396 23.2
## 4 Europe       360 21.1
## 5 Oceania       24  1.41
```

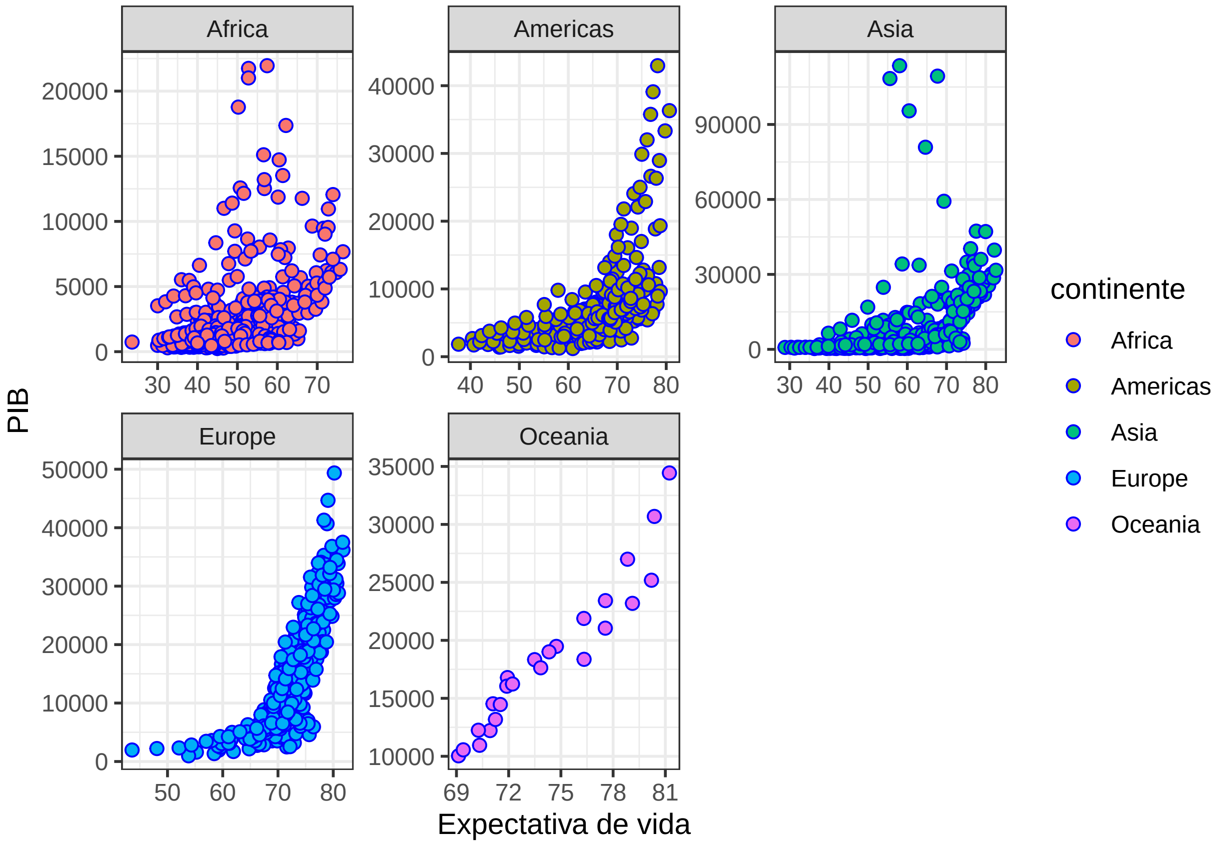
f. Gráfico de barras da tabela de frequência absoluta

```
barAbsF <- ggplot(data, aes(continente, absF)) +
  geom_bar(stat="identity", color="blue", fill="grey") +
  labs(x="Continent", y="Frequency") +
  theme_bw();
barAbsF;
```



g. Gráfico de dispersão entre expectativa de vida e pib per capita

```
dispPibExp <- ggplot(pib, aes(expVida, pibPercap, fill=continente)) +
  theme_bw() +
  labs(x="Expectativa de vida", y="PIB") +
  facet_wrap(~ continente, scale="free") +
  geom_point(size=2, shape=21, color="blue")
dispPibExp;
```

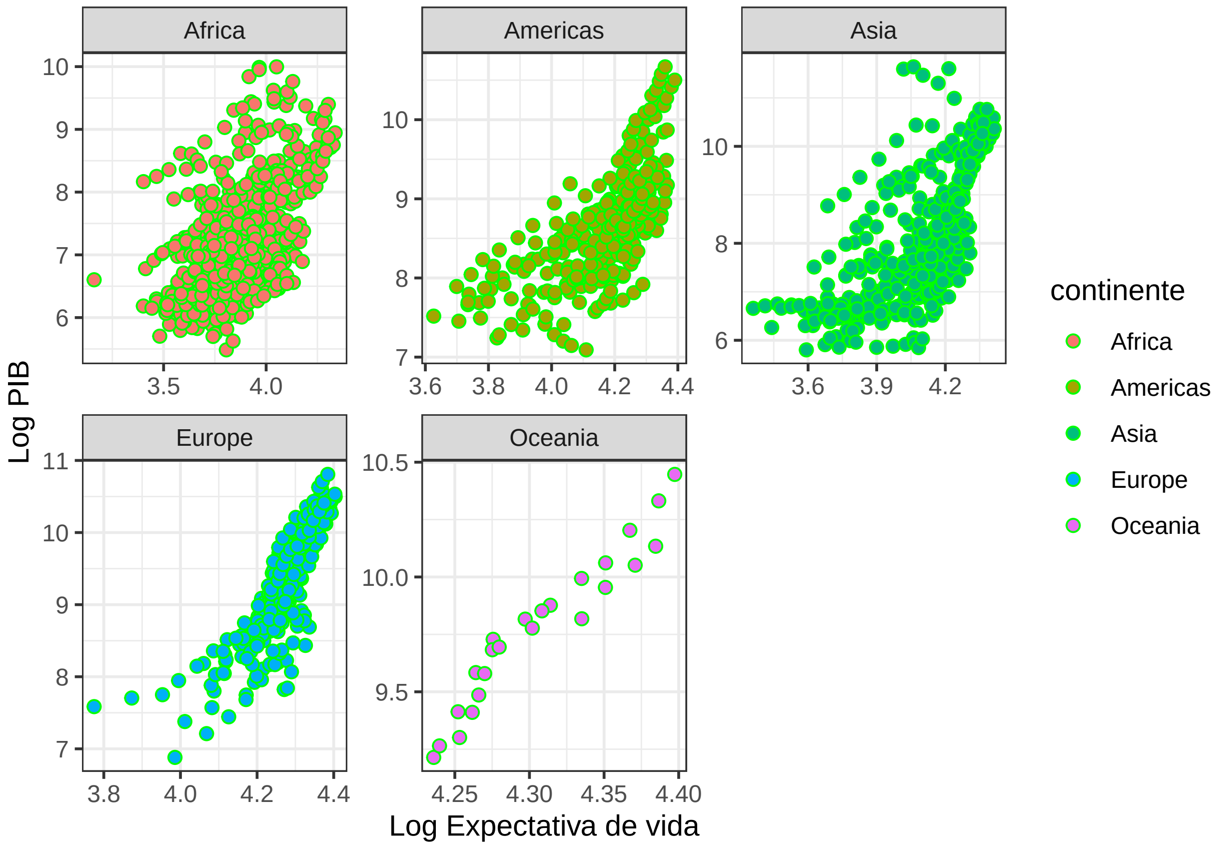


h. Adição de colunas logarítmicas a base de dados

```
pib$lpibPercap <- log(pib$pibPercap);
pib$lexpVida <- log(pib$expVida);
```

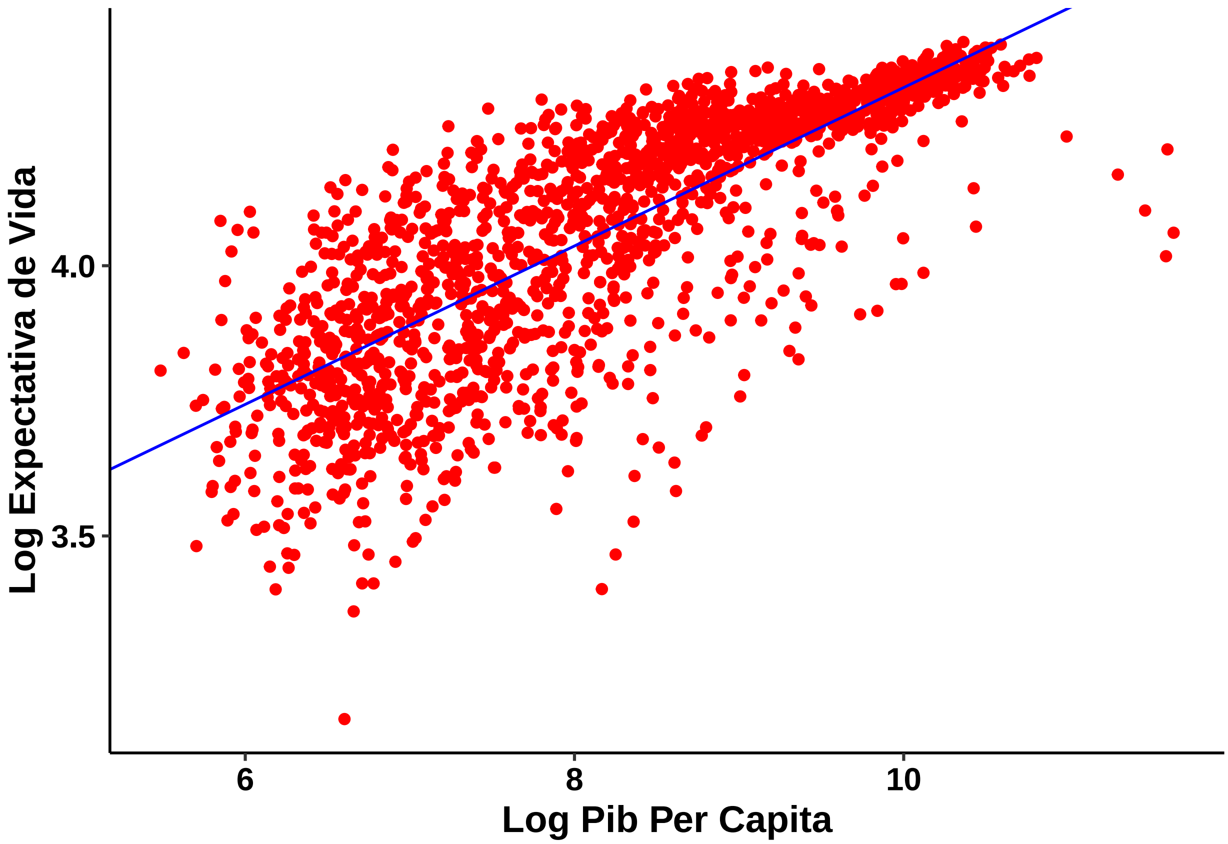
i. Gráfico de dispersão da relação das novas colunas

```
displPiblExp <- ggplot(pib, aes(lexpVida, lpibPercap, fill=continente)) +
  theme_bw() +
  labs(x="Log Expectativa de vida", y="Log PIB") +
  facet_wrap(~ continente, scale="free") +
  geom_point(size=2, shape=21, color="green");
displPiblExp;
```



j. Ajuste manual do modelo linear

```r
lMod <- ggplot(pib, aes(x=lpibPercap, y=lexpVida)) +
  geom_point(color = "red", fill = "blue") +
  xlab("Log Pib Per Capita") + ylab("Log Expectativa de Vida") +
  theme_pubr(legend = "right") +
  labs_pubr() +
  theme(plot.caption = element_text(hjust = 0));

sx <- sum(pib$lpibPercap);
ssx <- sum((pib$lpibPercap - mean(pib$lpibPercap))^2);

sy <- sum(pib$lexpVida);
ssy <- sum((pib$lexpVida - mean(pib$lexpVida))^2);

ssxy <- sum((pib$lpibPercap - mean(pib$lpibPercap)) * (pib$lexpVida - mean(pib$lexpVida)));

b1 <- ssxy / ssx;
b0 <- mean(pib$lexpVida) - b1 * mean(pib$lpibPercap);

lMod <- lMod + geom_abline(intercept=b0, slope=b1, color="blue");
lMod;
```

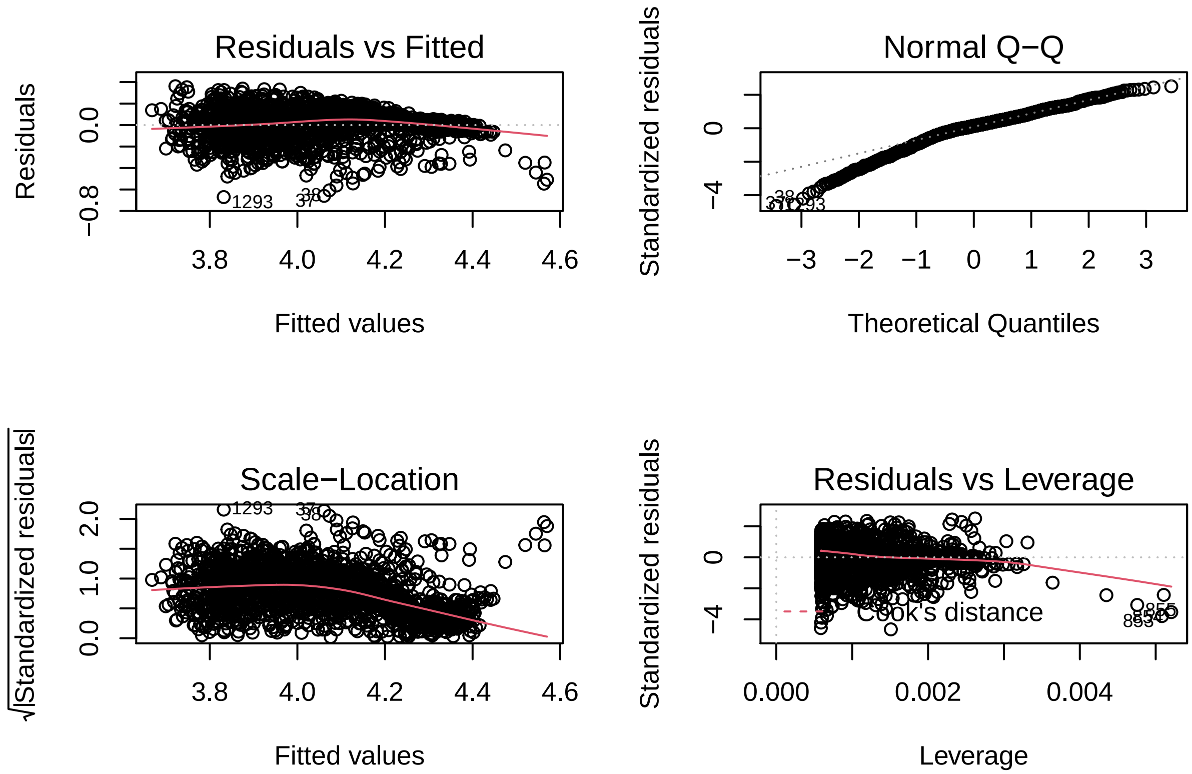


k. Análises de regressão

```
fit <- lm(lexpVida ~ lpibPercap, data = pib);
```

k.i Gráficos de diagnóstico

```
par(mfrow = c(2, 2))
plot(fit);
```



k.ii Avaliação do sumario do modelo

```
summary(fit);
```

```
##
## Call:
## lm(formula = lexpVida ~ lpibPercap, data = pib)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.67059 -0.06453  0.01978  0.09086  0.36156
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.864177   0.023283  123.02   <2e-16 ***
## lpibPercap  0.146549   0.002821   51.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1445 on 1702 degrees of freedom
## Multiple R-squared:  0.6132, Adjusted R-squared:  0.613
## F-statistic:  2698 on 1 and 1702 DF,  p-value: < 2.2e-16
```

k.iii Poder de explicação do modelo

```
shapiro.test(residuals(fit));
```

shapiro-wilk test para validação da normalização do modelo

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(fit)
## W = 0.96055, p-value < 2.2e-16
```

```
fit.df <- as.data.frame(residuals(fit));
names(fit.df) <- c("residuals");
aResiduals <- ggplot(fit.df, aes(residuals)) +
  geom_histogram(fill="blue", color="black", stat = "bin", bins = 10) +
  theme_pubr() + labs_pubr() + xlab("Residuals") + ylab("Frequency") +
  theme(plot.caption = element_text(hjust = 0));
aResiduals;
```

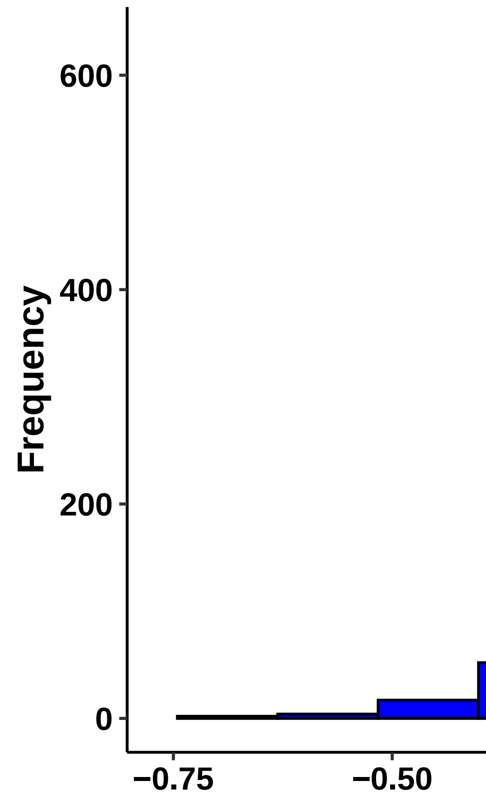


Gráfico de histograma dos resíduos do modelo, avaliando a normalização

Apesar da significância nas variáveis, os gráficos de diagnóstico e o teste shapiro-wilk mostram que não existe distribuição normal.

Problema 3

importando os dados de autos

```r
autos <- read.csv("autos.csv");
```

a.

```r
varDesc <- c(
  "Fabricante", "Tipo de combustível", "Combustão interna", "Número de portas", "Tipo de chassis", "Tra
);
varType <- c(1,1,1,1,1,1,1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0);
autosLinker <- build_linker(my.data=autos, variable_description=varDesc, variable_type=varType);
autosDict <- build_dict(
  my.data = autos,
  linker = autosLinker,
  option_description = NULL,
  prompt_varopts = FALSE
);

knitr::kable(autosDict, format='html', caption='');
```

variable_name

variable_description

variable_options

aspiration

Combustão interna

std

turbo

body.style

Tipo de chassis

convertible

hatchback

sedan

wagon

hardtop

bore

Diâmetro dos cilindros

2.54 to 3.94

city.mpg

Gasto médio de combustível na cidade

13 to 49

compression.ratio

Raio de compressão do pistão

7 to 23

curb.weight

Peso total

1488 to 4066

drive.wheels

Tração

rwd

fwd

4wd

engine.location

Localização do motor

front

rear

engine.size

Tamanho do motor

61 to 326

engine.type

Tipo de motor

dohc

ohcv

ohc

l

ohcf

fuel.system

Sistema de combustão

mpfi

2bbl

mfi

1bbl

spfi

idi

spdi

fuel.type

Tipo de combustível

gas

diesel

height

Altura

47.8 to 59.8

highway.mpg

Gasto médio de combustível em avenida

16 to 54

horsepower

Cavalos

48 to 262

length

Comprimento

141.1 to 208.1

make

Fabricante

alfa-romero

audi

bmw

chevrolet

dodge

honda

isuzu

jaguar

mazda

mercedes-benz

mercury

mitsubishi

nissan

peugot

plymouth

porsche

saab

subaru

toyota

volkswagen

volvo

num.cylinders

Quantidade de cilindros

3 to 12

num.doors

Número de portas

2

4

peak.rpm

Número máximo de RPM

4150 to 6600

price

Preço

5118 to 45400

stroke

Tipo de pistão

2.07 to 4.17

wheel.base

Base de rodas

86.6 to 120.9

width

Largura

60.3 to 72

b.

```
str(autos);
```

```
## 'data.frame':    193 obs. of  24 variables:
##  $ make             : chr  "alfa-romero" "alfa-romero" "alfa-romero" "audi" ...
##  $ fuel.type        : chr  "gas" "gas" "gas" "gas" ...
##  $ aspiration       : chr  "std" "std" "std" "std" ...
##  $ num.doors        : int  2 2 2 4 4 2 4 4 4 2 ...
##  $ body.style       : chr  "convertible" "convertible" "hatchback" "sedan" ...
##  $ drive.wheels     : chr  "rwd" "rwd" "rwd" "fwd" ...
##  $ engine.location  : chr  "front" "front" "front" "front" ...
##  $ wheel.base       : num  88.6 88.6 94.5 99.8 99.4 ...
##  $ length           : num  169 169 171 177 177 ...
##  $ width            : num  64.1 64.1 65.5 66.2 66.4 66.3 71.4 71.4 71.4 64.8 ...
```

```
##  $ height           : num  48.8 48.8 52.4 54.3 54.3 53.1 55.7 55.7 55.9 54.3 ...
##  $ curb.weight      : int  2548 2548 2823 2337 2824 2507 2844 2954 3086 2395 ...
##  $ engine.type      : chr  "dohc" "dohc" "ohcv" "ohc" ...
##  $ num.cylinders    : int  4 4 6 4 5 5 5 5 5 4 ...
##  $ engine.size      : int  130 130 152 109 136 136 136 136 131 108 ...
##  $ fuel.system      : chr  "mpfi" "mpfi" "mpfi" "mpfi" ...
##  $ bore             : num  3.47 3.47 2.68 3.19 3.19 3.19 3.19 3.19 3.13 3.5 ...
##  $ stroke           : num  2.68 2.68 3.47 3.4 3.4 3.4 3.4 3.4 3.4 2.8 ...
##  $ compression.ratio: num  9 9 9 10 8 8.5 8.5 8.5 8.3 8.8 ...
##  $ horsepower       : int  111 111 154 102 115 110 110 110 140 101 ...
##  $ peak.rpm         : int  5000 5000 5000 5500 5500 5500 5500 5500 5500 5800 ...
##  $ city.mpg         : int  21 21 19 24 18 19 19 19 17 23 ...
##  $ highway.mpg      : int  27 27 26 30 22 25 25 25 20 29 ...
##  $ price            : int  13495 16500 16500 13950 17450 15250 17710 18920 23875 16430 ...
```

c.

c.i

```
summary(autos);
```

```
##      make             fuel.type          aspiration          num.doors
##  Length:193         Length:193         Length:193         Min.   :2.000
##  Class :character   Class :character   Class :character   1st Qu.:2.000
##  Mode  :character   Mode  :character   Mode  :character   Median :4.000
##                                                           Mean   :3.161
##                                                           3rd Qu.:4.000
##                                                           Max.   :4.000
##   body.style        drive.wheels       engine.location      wheel.base
##  Length:193         Length:193         Length:193         Min.   : 86.60
##  Class :character   Class :character   Class :character   1st Qu.: 94.50
##  Mode  :character   Mode  :character   Mode  :character   Median : 97.00
##                                                           Mean   : 98.92
##                                                           3rd Qu.:102.40
##                                                           Max.   :120.90
##      length          width           height       curb.weight
##  Min.   :141.1   Min.   :60.30   Min.   :47.80   Min.   :1488
##  1st Qu.:166.3   1st Qu.:64.10   1st Qu.:52.00   1st Qu.:2145
##  Median :173.2   Median :65.40   Median :54.10   Median :2414
##  Mean   :174.3   Mean   :65.89   Mean   :53.87   Mean   :2562
##  3rd Qu.:184.6   3rd Qu.:66.90   3rd Qu.:55.70   3rd Qu.:2952
##  Max.   :208.1   Max.   :72.00   Max.   :59.80   Max.   :4066
##  engine.type        num.cylinders    engine.size    fuel.system
##  Length:193         Min.   : 3.00   Min.   : 61.0   Length:193
##  Class :character   1st Qu.: 4.00   1st Qu.: 98.0   Class :character
##  Mode  :character   Median : 4.00   Median :120.0   Mode  :character
##                     Mean   : 4.42   Mean   :128.1
##                     3rd Qu.: 4.00   3rd Qu.:146.0
##                     Max.   :12.00   Max.   :326.0
##       bore           stroke      compression.ratio   horsepower
##  Min.   :2.540   Min.   :2.070   Min.   : 7.00     Min.   : 48.0
##  1st Qu.:3.150   1st Qu.:3.110   1st Qu.: 8.50     1st Qu.: 70.0
```

```
##  Median :3.310   Median :3.290   Median : 9.00     Median : 95.0
##  Mean   :3.331   Mean   :3.249   Mean   :10.14     Mean   :103.5
##  3rd Qu.:3.590   3rd Qu.:3.410   3rd Qu.: 9.40     3rd Qu.:116.0
##  Max.   :3.940   Max.   :4.170   Max.   :23.00     Max.   :262.0
##     peak.rpm        city.mpg      highway.mpg        price
##  Min.   :4150   Min.   :13.00   Min.   :16.00   Min.   : 5118
##  1st Qu.:4800   1st Qu.:19.00   1st Qu.:25.00   1st Qu.: 7738
##  Median :5100   Median :25.00   Median :30.00   Median :10245
##  Mean   :5100   Mean   :25.33   Mean   :30.79   Mean   :13285
##  3rd Qu.:5500   3rd Qu.:30.00   3rd Qu.:34.00   3rd Qu.:16515
##  Max.   :6600   Max.   :49.00   Max.   :54.00   Max.   :45400
```

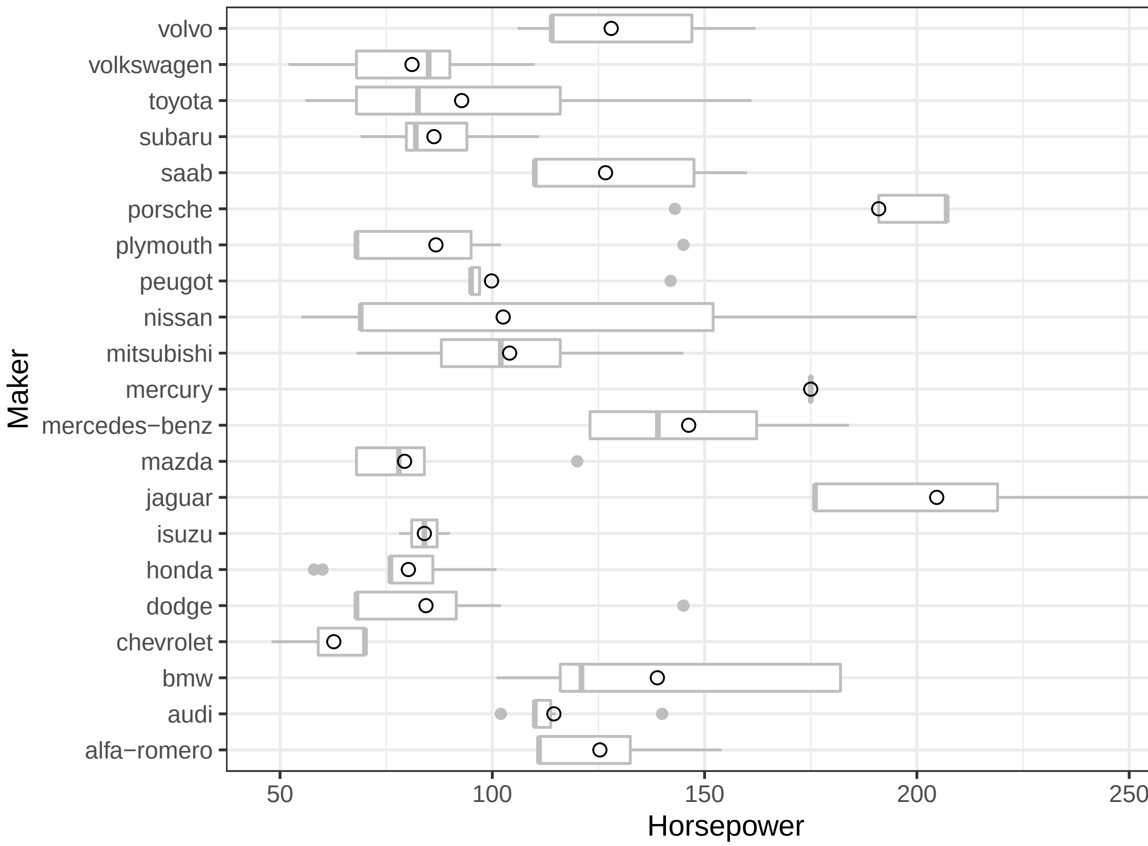
c.ii

```
table(autos$make);
```

```
##
##   alfa-romero          audi           bmw     chevrolet         dodge
##             3             6             8             3             8
##         honda         isuzu        jaguar         mazda mercedes-benz
##            13             2             3            12             8
##       mercury    mitsubishi        nissan        peugot      plymouth
##             1            13            18            11             7
##       porsche          saab        subaru        toyota    volkswagen
##             4             6            12            32            12
##         volvo
##            11
```

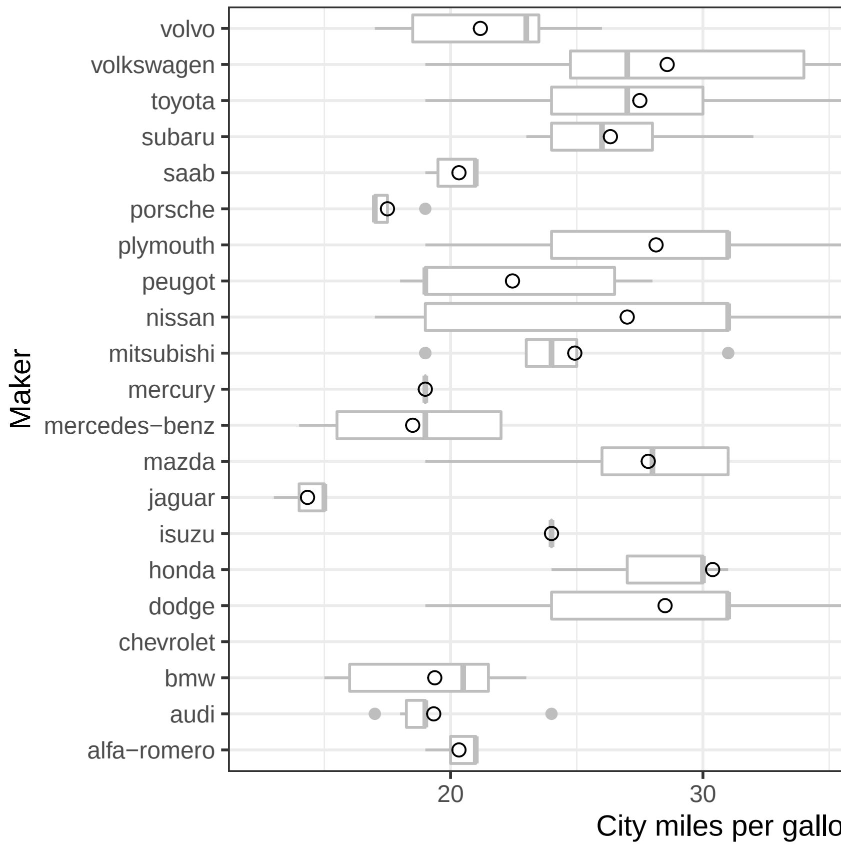
c.iii

```
hpMK <- ggplot(autos, aes(horsepower, make)) +
  theme_bw() +
  geom_boxplot(color="grey") +
  labs(x="Horsepower", y="Maker") +
  stat_summary(fun="mean", geom="point", shape=1, size=2, color="black");
hpMK;
```



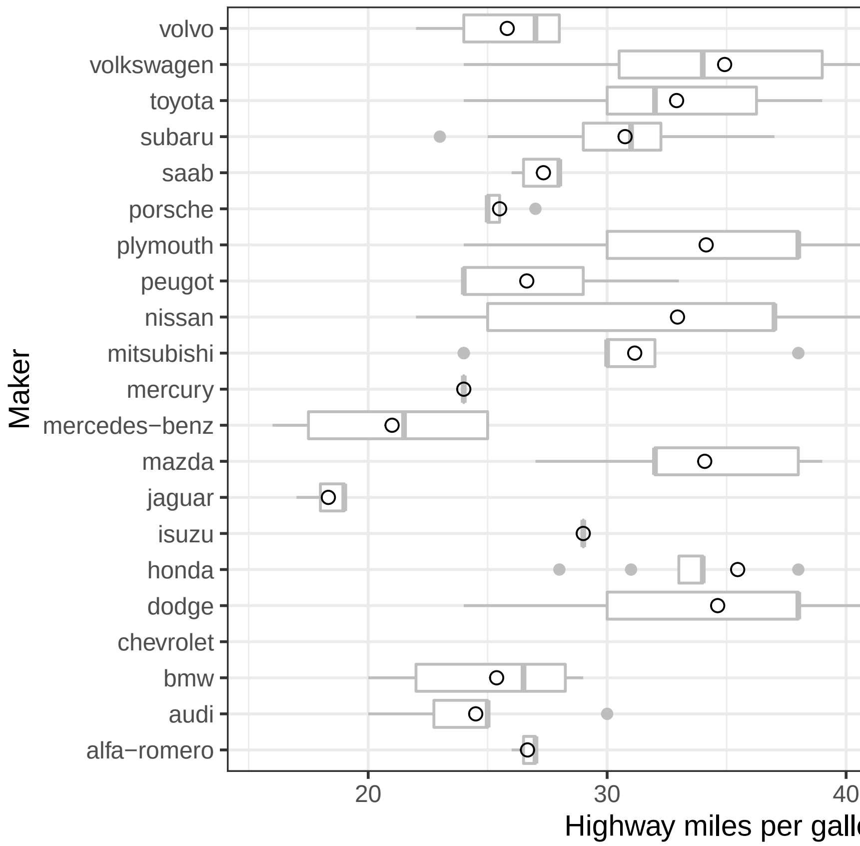
Cavalos x Fabricante

```
cmMK <- ggplot(autos, aes(city.mpg, make)) +
  theme_bw() +
  geom_boxplot(color="grey") +
  labs(x="City miles per gallon", y="Maker") +
  stat_summary(fun="mean", geom="point", shape=1, size=2, color="black");
cmMK;
```



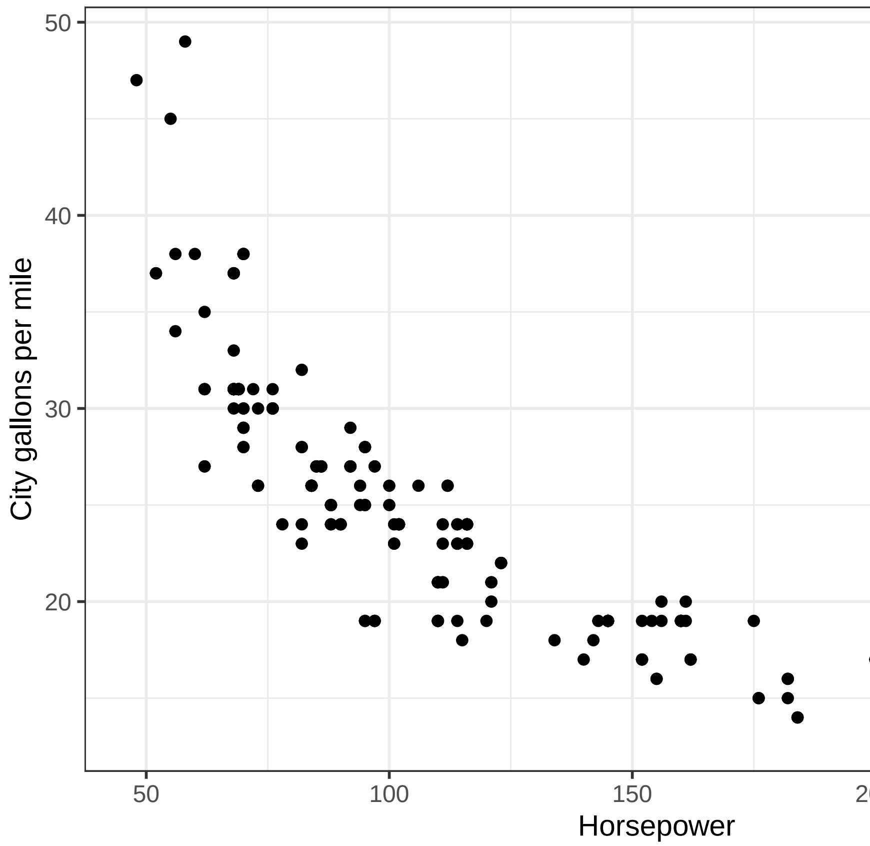
Gasto de combustível na cidade x Fabricante

```
hmMK <- ggplot(autos, aes(highway.mpg, make)) +
  theme_bw() +
  geom_boxplot(color="grey") +
  labs(x="Highway miles per gallon", y="Maker") +
  stat_summary(fun="mean", geom="point", shape=1, size=2, color="black");
hmMK;
```



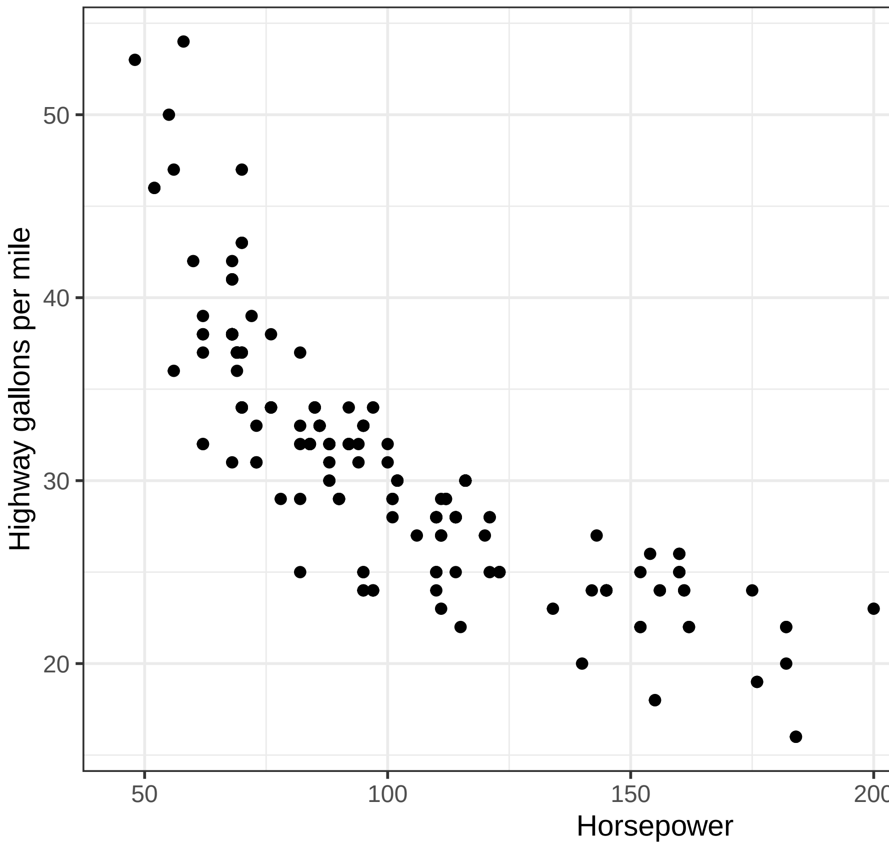
Gasto de combustível na pista x Fabricante

```
hpCM <- ggplot(autos, aes(horsepower, city.mpg)) +
  theme_bw() +
  geom_point() +
  labs(x="Horsepower", y="City gallons per mile");
hpCM;
```



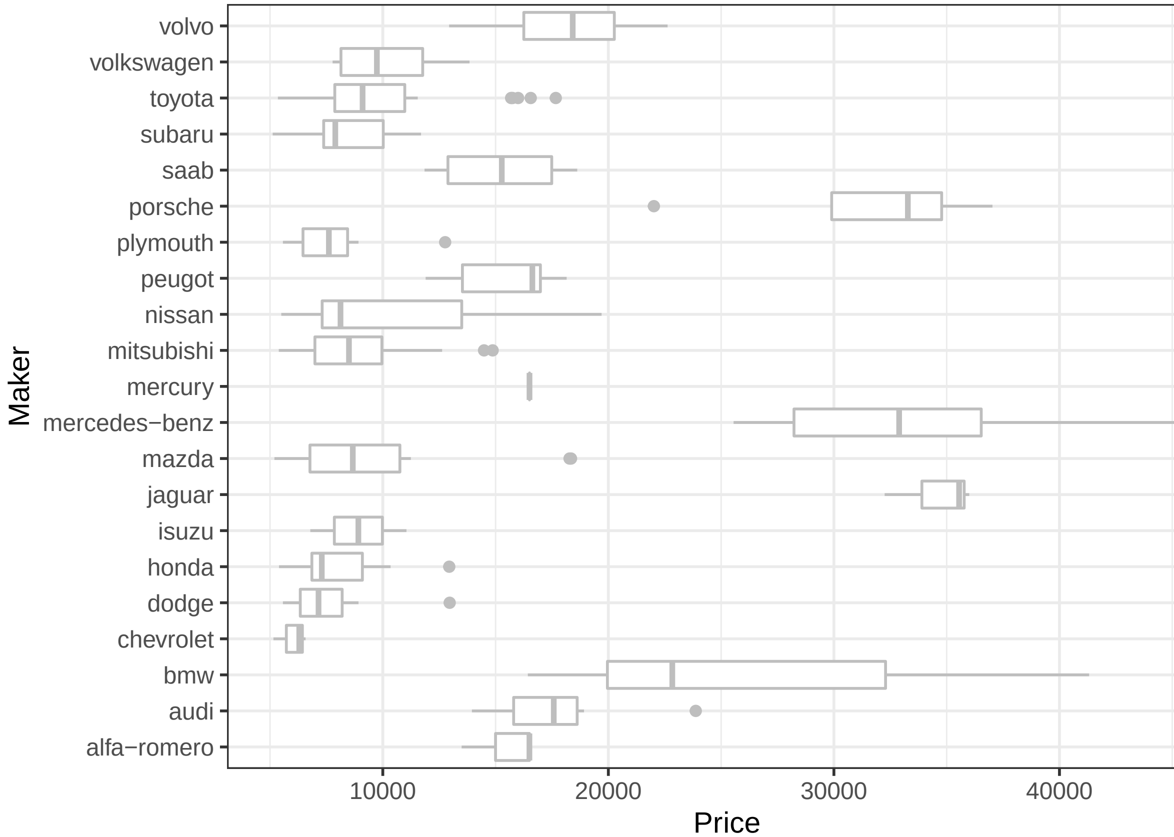
Cavalos x Gasto de combustível na cidade

```
hpHM <- ggplot(autos, aes(horsepower, highway.mpg)) +
  theme_bw() +
  geom_point() +
  labs(x="Horsepower", y="Highway gallons per mile");
hpHM;
```



Cavalos x Gasto de combustível na pista

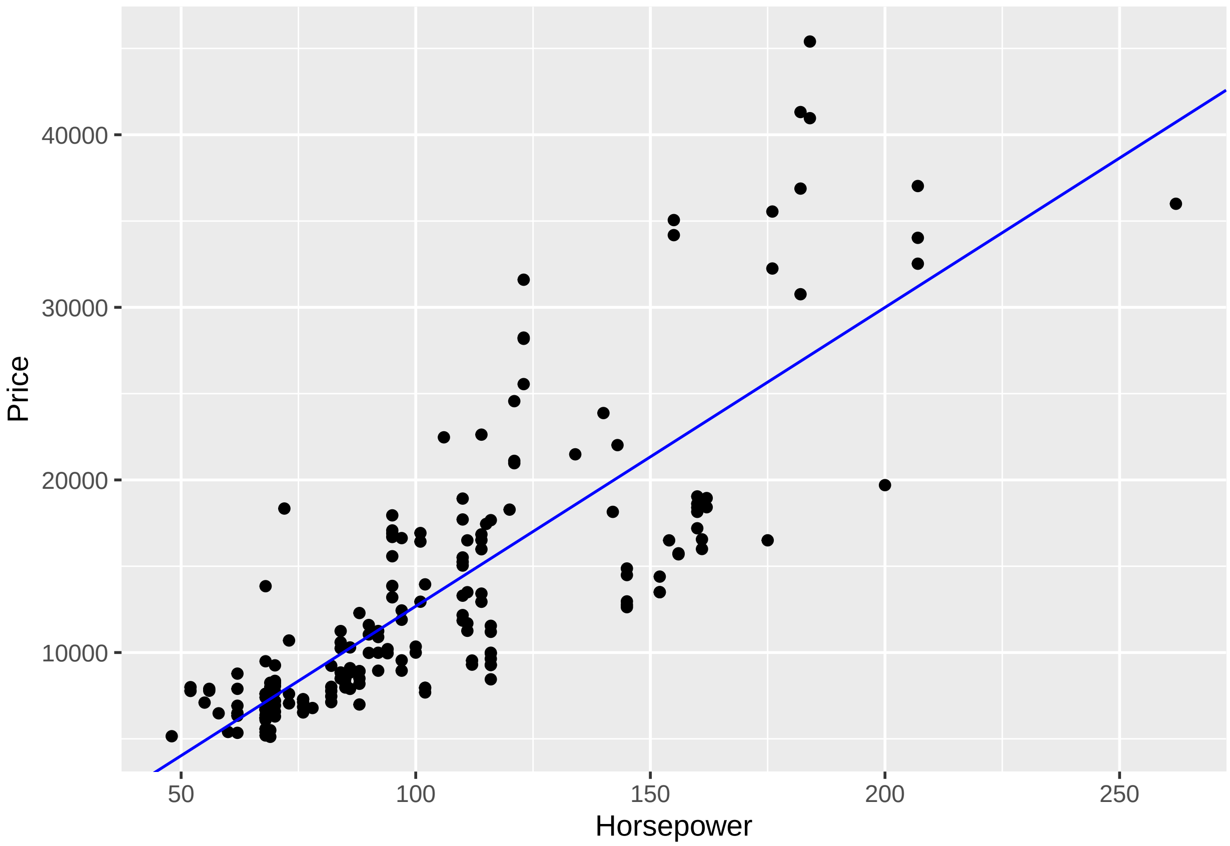
```
priceMk <- ggplot(autos, aes(price, make)) +
  theme_bw() +
  geom_boxplot(color="grey") +
  labs(x="Price", y="Maker");
priceMk;
```



Preço x Fabricante

d. Regressão linear da potência com os preços

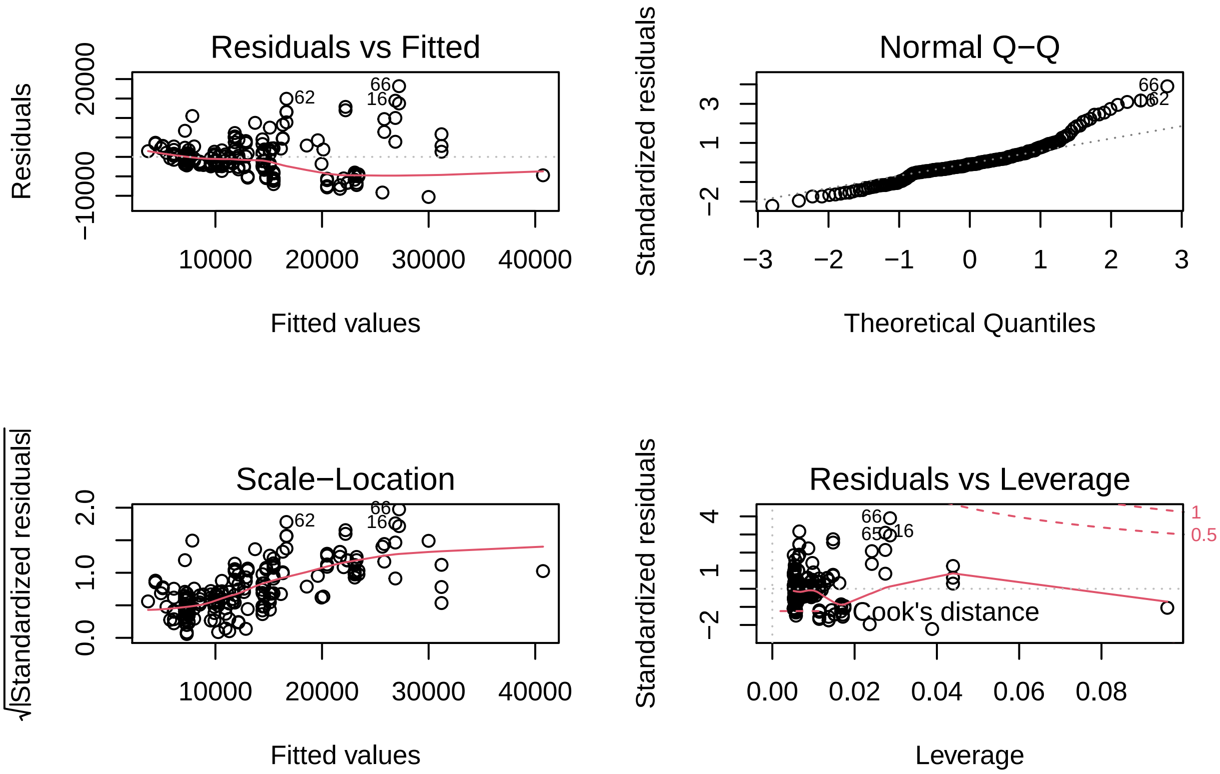
```
fitPriceHP <- lm(price ~ horsepower, data=autos);
cofsFPHP <- coef(fitPriceHP);
hpPrice <- ggplot(autos, aes(horsepower, price)) +
  geom_point() +
  labs(x="Horsepower", y="Price");

hpPrice <- hpPrice + geom_abline(intercept=cofsFPHP[1], slope=cofsFPHP[2], color="blue");
hpPrice;
```



e. Análises da regressão

e.i Gráficos de diagnóstico

```
par(mfrow = c(2, 2))
plot(fitPriceHP);
```




```
fitPriceHP.df <- as.data.frame(residuals(fitPriceHP));
names(fitPriceHP.df) <- c("residuals");
aResiduals <- ggplot(fitPriceHP.df, aes(residuals)) +
  geom_histogram(fill="blue", color="black", stat = "bin", bins = 10) +
  theme_pubr() + labs_pubr() + xlab("Residuals") + ylab("Frequency") +
  theme(plot.caption = element_text(hjust = 0));
aResiduals;
```

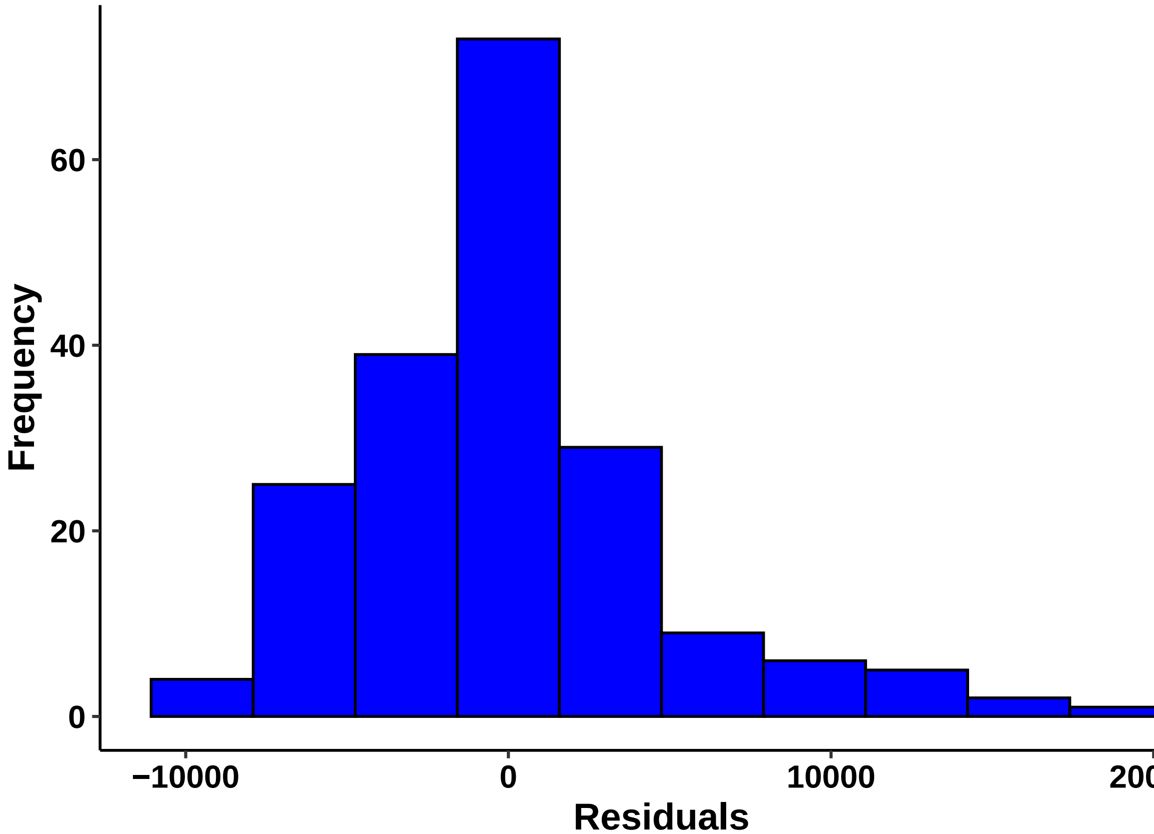


Gráfico dos resíduos

e.ii Sumário da regressão

```
summary(fitPriceHP)
```

```
##
## Call:
## lm(formula = price ~ horsepower, data = autos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10296.1  -2243.5   -450.1   1794.7  18174.9
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4630.70     990.58  -4.675 5.55e-06 ***
## horsepower    173.13       8.99  19.259  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4728 on 191 degrees of freedom
## Multiple R-squared:  0.6601, Adjusted R-squared:  0.6583
## F-statistic: 370.9 on 1 and 191 DF,  p-value: < 2.2e-16
```

f. Resultados do ajuste

f.i

O intercept é -4630.70, valor incoerente dado ao tipo da variável preço.

```
fitPriceHP;
```

```
##
## Call:
## lm(formula = price ~ horsepower, data = autos)
##
## Coefficients:
## (Intercept)   horsepower
##     -4630.7        173.1
```

```
gvlma(fitPriceHP)
```

```
##
## Call:
## lm(formula = price ~ horsepower, data = autos)
##
## Coefficients:
## (Intercept)   horsepower
##     -4630.7        173.1
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = fitPriceHP)
##
##                      Value   p-value                   Decision
## Global Stat        64.7104 2.961e-13 Assumptions NOT satisfied!
## Skewness           30.8252 2.823e-08 Assumptions NOT satisfied!
## Kurtosis           26.5687 2.543e-07 Assumptions NOT satisfied!
## Link Function       0.9265 3.358e-01    Assumptions acceptable.
## Heteroscedasticity  6.3899 1.148e-02 Assumptions NOT satisfied!
```

f.ii

O modelo pode ser corrigido pela aplicação do log na variável explicativa (price).

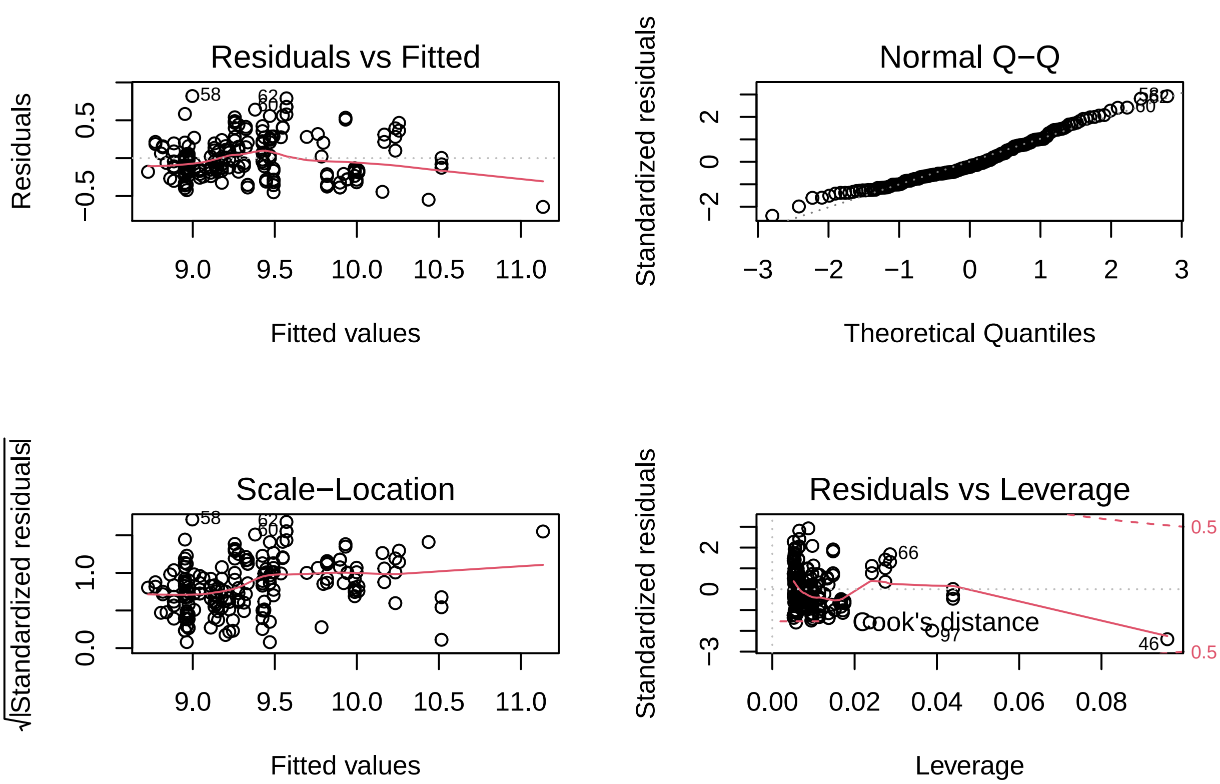
```
logFitHP <- lm(log(price) ~ horsepower, data=autos)
logFitHP
```

```
##
## Call:
## lm(formula = log(price) ~ horsepower, data = autos)
##
## Coefficients:
## (Intercept)   horsepower
##     8.18729      0.01125
```

```
gvlma(logFitHP)
```

```
##
## Call:
## lm(formula = log(price) ~ horsepower, data = autos)
##
## Coefficients:
## (Intercept)   horsepower
##     8.18729      0.01125
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = logFitHP)
##
##                         Value   p-value                   Decision
## Global Stat          27.04624 1.946e-05 Assumptions NOT satisfied!
## Skewness             11.65833 6.392e-04 Assumptions NOT satisfied!
## Kurtosis              0.07493 7.843e-01    Assumptions acceptable.
## Link Function        11.47897 7.039e-04 Assumptions NOT satisfied!
## Heteroscedasticity  3.83401 5.022e-02    Assumptions acceptable.
```

```
par(mfrow = c(2, 2))
plot(logFitHP)
```



```r
summary(logFitHP)
```

```
##
## Call:
## lm(formula = log(price) ~ horsepower, data = autos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64356 -0.18862 -0.06348  0.19537  0.81976
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8.1872907  0.0589965  138.78   <2e-16 ***
## horsepower  0.0112502  0.0005354   21.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2816 on 191 degrees of freedom
## Multiple R-squared:  0.698,  Adjusted R-squared:  0.6965
## F-statistic: 441.5 on 1 and 191 DF,  p-value: < 2.2e-16
```

É possível ver uma leve melhora no R^2 ajustado, melhorando a explicação do modelo.

g. A influência da potência no preço

Quanto maior a potência, mais caro o carro, isso é visível em como a regressão fica e seguindo a faixa de erro do Normal Q-Q, onde os erros são mais aparentes apenas no fim da distribuição, o p-value não mostra um valor significativa para a hipótese nula, logo é possível afirmar que as variáveis possuem relação, apesar do R quadrado ajustado paresentar um valor um tanto quanto baixo. Contudo, a relação do preço com a potência, seguindo o resultado de 0.65 do R^2 ajustado, ainda mostra que outras variáveis podem influenciar no preço final.