Data Mining
Análise por Regressão

Prof. Dr. Joaquim Assunção

DEPARTAMENTO DE COMPUTAÇÃO APLICADA
CENTRO DE TECNOLOGIA
UFSM
2019

www.inf.ufsm.br/~joaquim

Fair user agreement

Este material foi criado para a disciplina de Mineração de Dados - Centro de Tecnologia da UFSM.

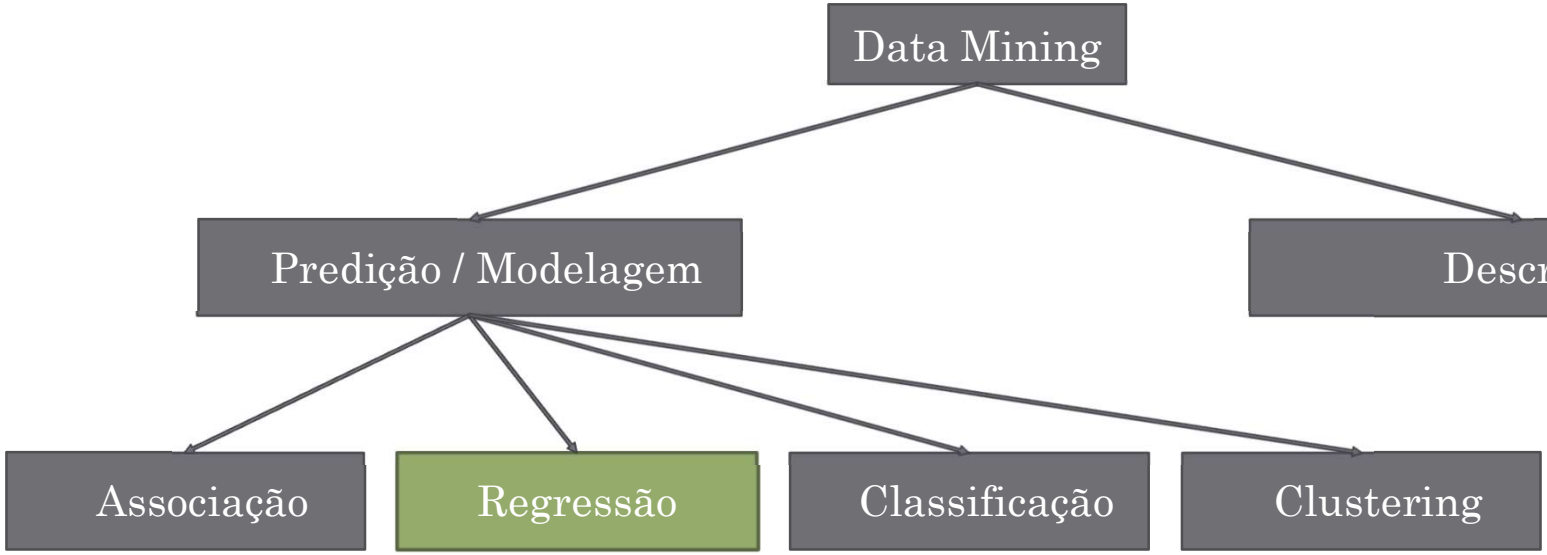
Você pode usar este material livremente*; porém, caso seja usado em outra instituição, **me envie um e-mail** avisando o nome da instituição e a disciplina.

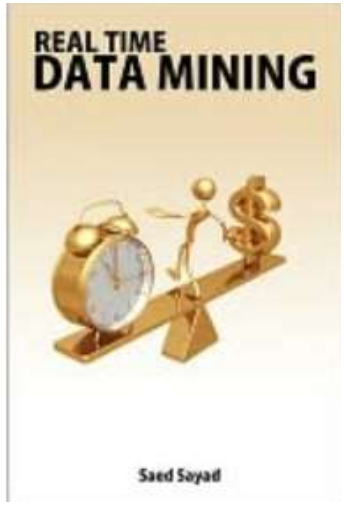
*Caso você queira usar algo desse material em alguma publicação, envie-me um e-mail com antecedência.

Prof. Dr. Joaquim Assunção.

joaquim@inf.ufsm.br

Mapa para Mineração de Dados*





***http://www.saedsayad.com/data_mining_map.htm**

Definições

- “Regressão pode ser vista como uma técnica de modelagem preditiva onde a variável alvo a ser avaliada é contínua.”*
- Exemplos incluem a previsão de índice de bolsa de valores com base em fatores econômicos ou chuvas em uma região com base em demais características meteorológicas.

*Por Tan et al. 2016

- ...Em termos práticos, regressão e classificação são tão parecidos que são usados de maneira intercambiável.

Definições

- Em geral, algoritmos de regressão são usados para prever o valor de uma variável de resposta (dependente) de uma ou mais variáveis preditoras (independentes).
- Existem várias formas de regressão; tais como, linear, múltipla, ponderada, polinomial, não paramétrica e robusta.

Formalmente

- Suponha que D denote um conjunto de dados que contenha N observações.

$$D = \{(x_i, y_i) | \, i = 1,2, \ldots, N\}$$

Cada x_i corresponde a um atributo de uma variável explicativa e y_i corresponde a variável alvo. Os atributos podem ser discretos ou contínuos.

...

- Em linhas gerais, podemos dizer que o objetivo da regressão é encontrar uma função com o menor erro possível em relação aos dados. ... Para isso temos 2 tipos de erro.

$$\sum_{i}^{N} |y_i - f(x_i)|$$ Erro absoluto

$$\sum_{i}^{N} (y_i - f(x_i))^2$$ Erro Quadrático ou SSE (*sum of squared error*)

Exemplo – *Old Faithful Geyser*

- Situado no parque nacional de Yellowstone, USA.
- Possui uma característica geotérmica altamente previsível, e entra em erupção em intervalos de 44 a 125 minutos desde o ano 2000!

Dados → equação para previsão

```
> data(faithful)
> head(faithful)
  eruptions waiting
1     3.600      79
2     1.800      54
3     3.333      74
4     2.283      62
5     4.533      85
6     2.883      55
```

$$\hat{DE} = \hat{b}_0 + \hat{b}_1 TE$$

- O ponto de intersecção e o ponto de espera (constantes).
- Dado o novo tempo de espera (TE), a Duração estimada (DE) é obtida.

```
> nrow(faithful)
[1] 272
```

Também poderíamos plugar uma variável para o erro esperado.

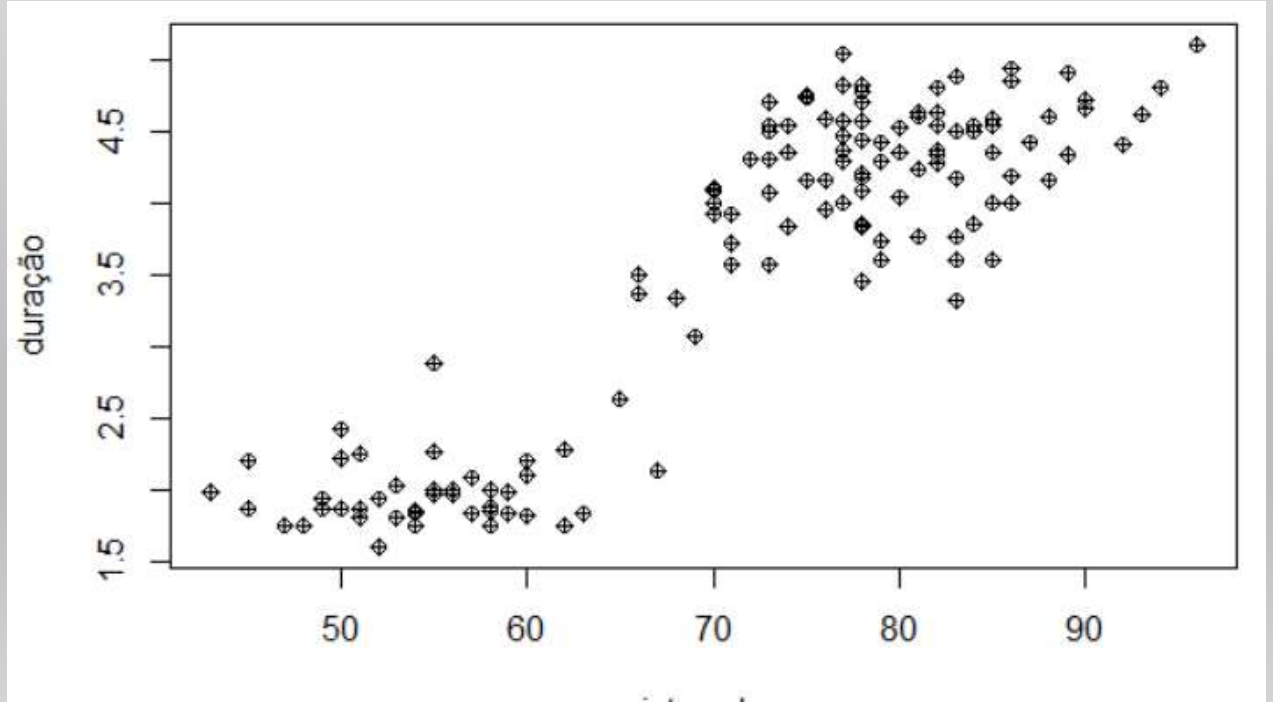
Dados → partição

- A partição deve ser feita para treino e teste. Geralmente 2/3 e 1/3. Para simplificar, vamos usar a função `createDataPartition` com 0.5

```
> inTrain <- createDataPartition(y=faithful$waiting,
+                                p=0.5, list=FALSE)
> trainFaith <- faithful[inTrain,];
> testFaith <- faithful[-inTrain,]
> nrow(trainFaith)
[1] 137
> nrow(testFaith)
[1] 135
```

Dados → visualização

- Os dados podem ser visualizados via `plot()`. Neste caso estamos visualizando os dados de treino. (`trainFaith$waiting,trainFaith$eruptions`)



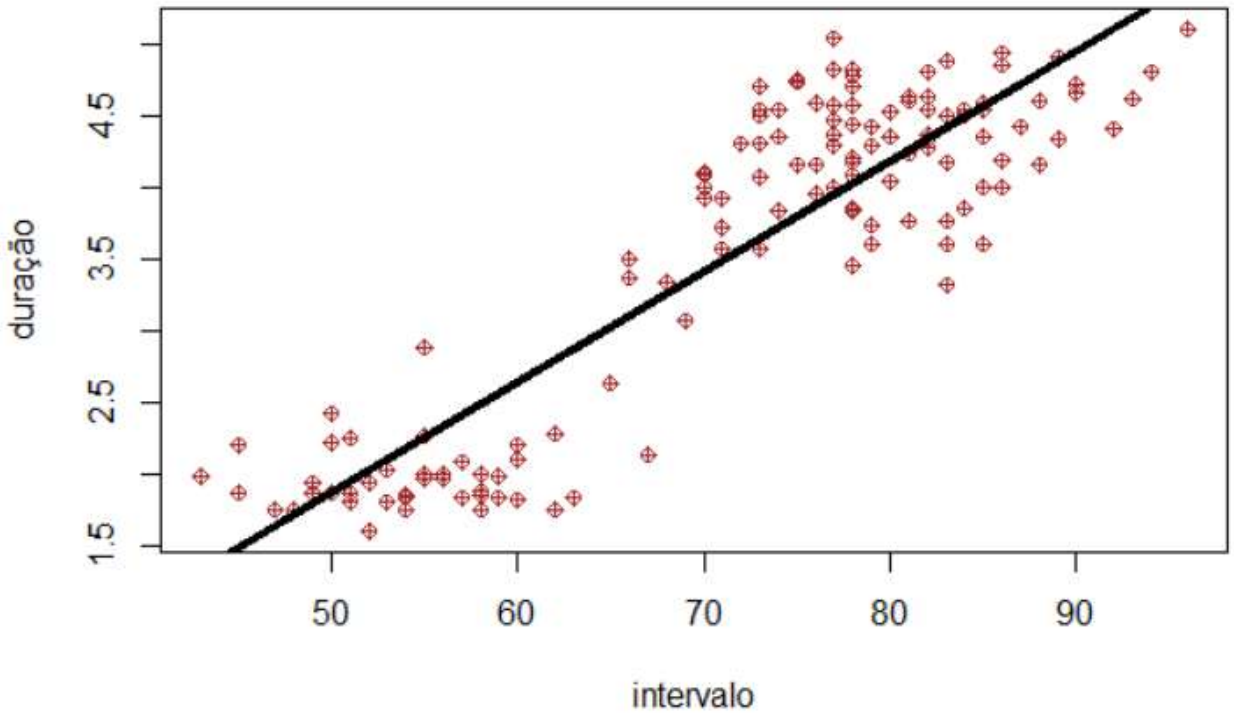
Ajuste do modelo

- Agora podemos usar a partição de treino para ajustar o modelo. Podemos usar a função `lm` para um modelo linear.
- O primeiro parâmetro é a variável a ser predita e a variável preditora, respectivamente, e separados por `~`.

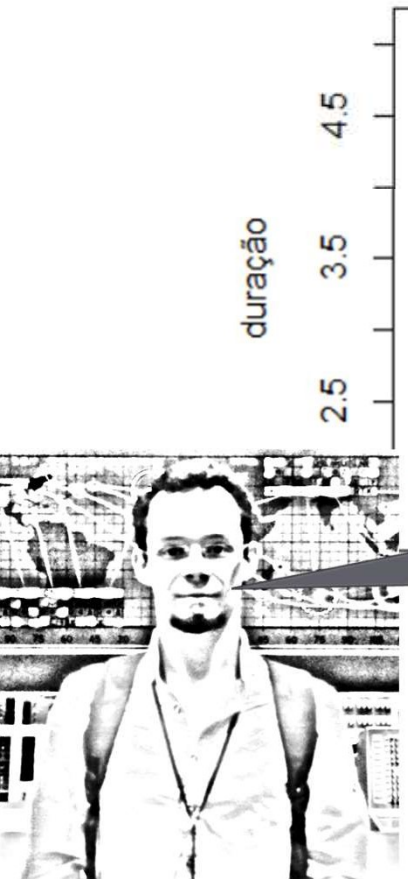
Ajuste do modelo

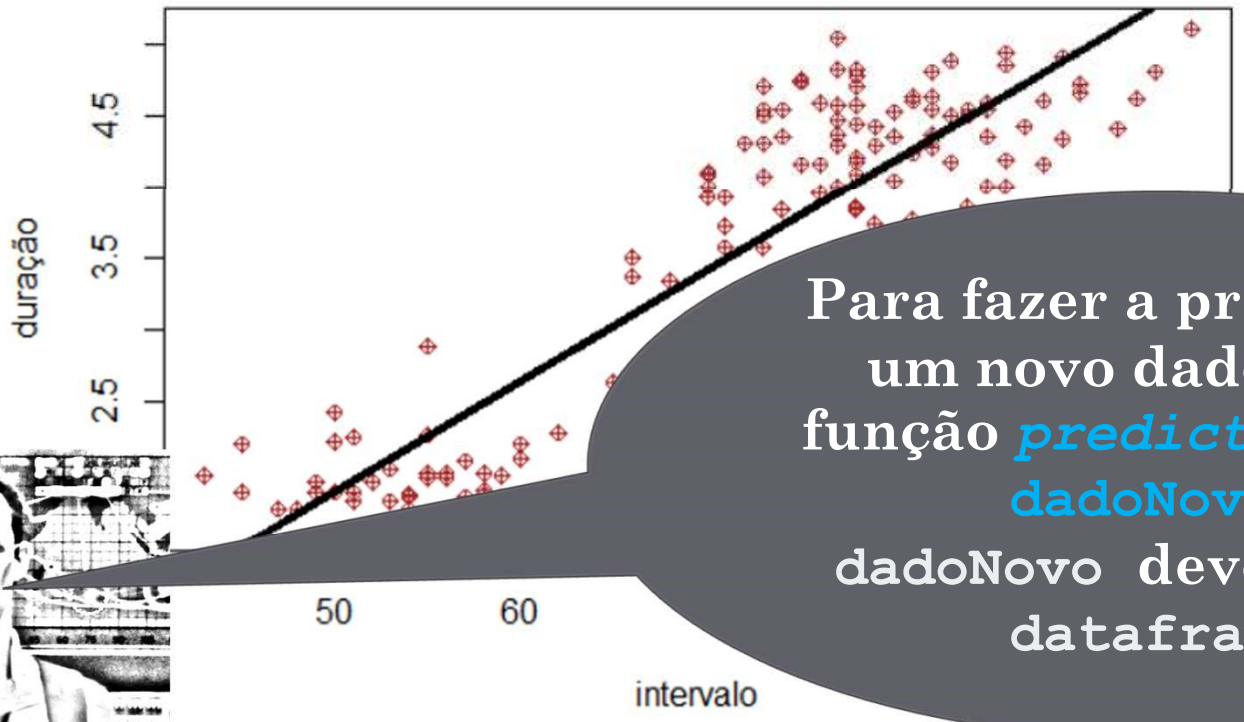
```
> meuModeloLinear <- lm(eruptions ~ waiting,data=trainFaith)
> summary(meuModeloLinear)

Call:
lm(formula = eruptions ~ waiting, data = trainFaith)

Residuals:
     Min       1Q   Median       3Q      Max
-1.10709 -0.38364 -0.00828  0.38551  1.20132

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.843375   0.228775  -8.058 3.66e-13 ***
waiting      0.075119   0.003178  23.636  < 2e-16 ***
```

Plot



Predict





Para fazer a predição de um novo dado, use a função `predict(modelo, dadoNovo)`

`dadoNovo` deve ser um `dataframe`

…A wild Geyser appears. Hands On!

1. Abra o arquivo "*GeyserUFSM.csv*". Use duas partições, treino e teste. Crie um modelo linear e faça a predição para os seguintes tempos de espera: 200, 230, 245, e 270.

Exemplo II

- Nosso objetivo é comparar o consumo de carros com cambio manual vs carros com cambio automático.
  - Para isso, primeiramente vamos carregar um dataset com este tipo de dado no R.
  - Vamos usar o *mtcars* que já vem na instalação base do R.
  - A função `data()` carrega o dataset.

Exemplo

- Eis uma amostra de *mtcars*

```
> mtcars
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
```

am é nosso alvo.
Mpg *(*milhas por galão*) e* ***wt*** *(*peso*) são nossas variáveis preditoras*

Exemplo

- A conversão abaixo é feita para trocar as unidades imperiais para algo mais familiar.

```
data(mtcars)
mtcars$kpl <- mtcars$mpg * 0.425
mtcars$peso <- mtcars$wt * 0.453
```

Exemplo

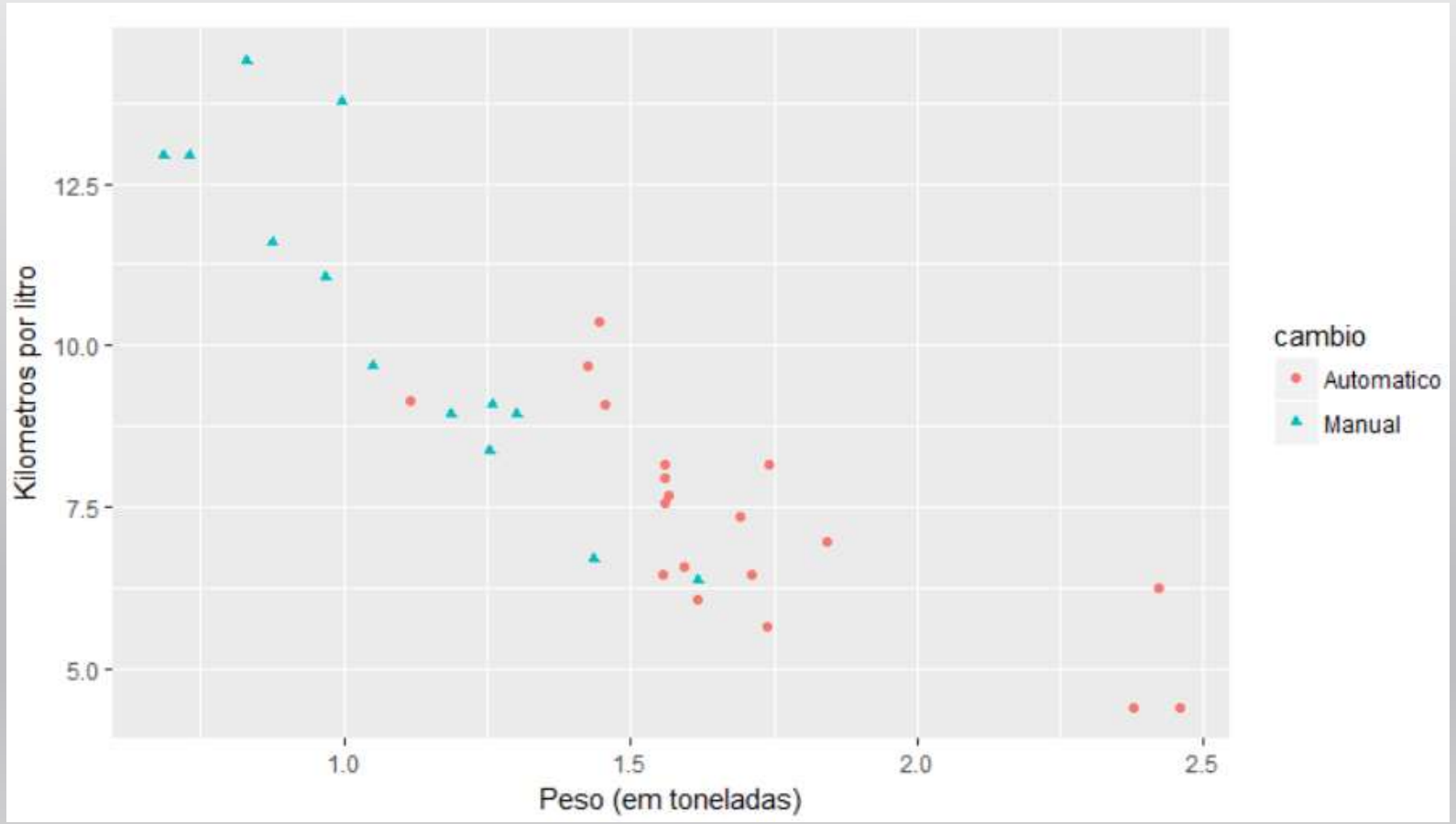
- Após, vamos separar os dados na variável *cambio*. Usaremos o tipo *factor* para separar os dados em automático e manual (atributo ***am*** do *dataset*).
- O próximo passo é criar um modelo linear entre peso e quilômetros por litro.
- Em R, usamos a funçao **lm** que tem como parâmetros "(*formula*, *data*, ...)".

Exemplo

```
cambio <- factor(mtcars$am, levels = c(0,1), labels=c("Automatico","Manual"))
modelo <- lm(peso~kpl, data=mtcars)
```

Já temos um modelo linear. Mas antes vamos ver como estão nossos dados...

Exemplo

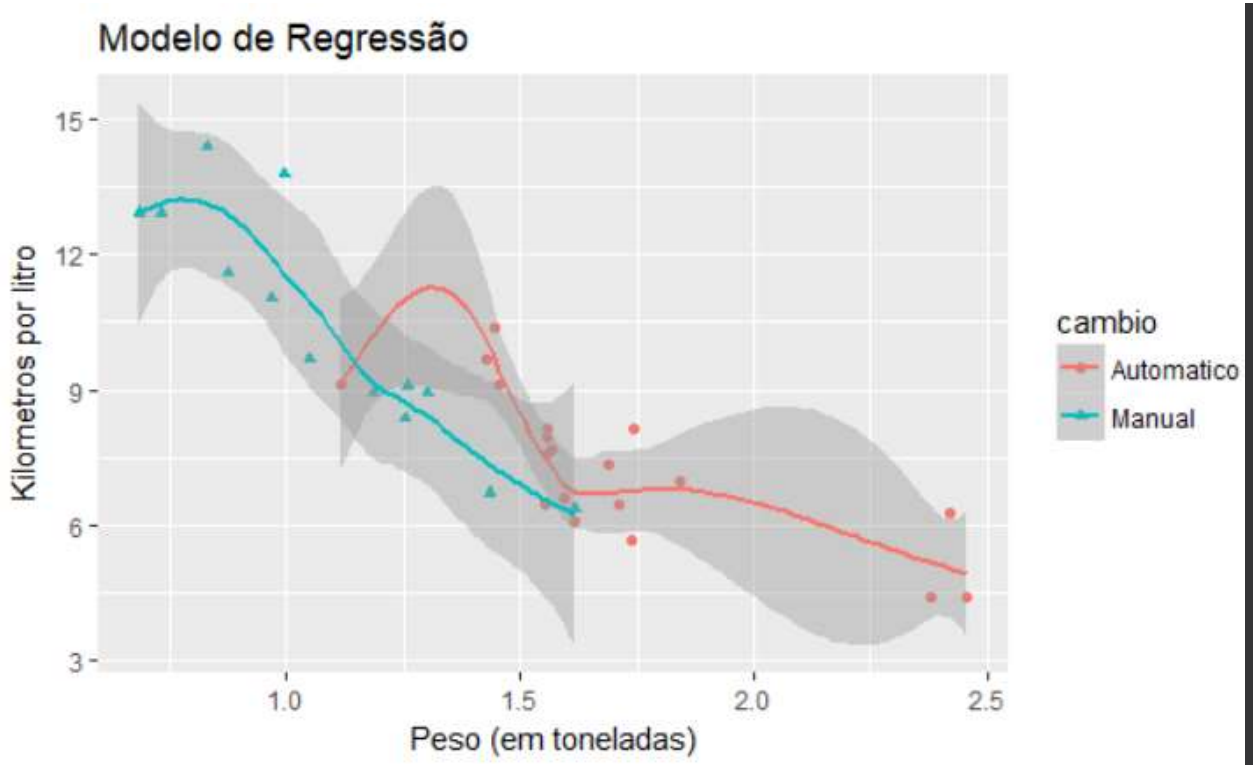

Kilometros por litro
12.5
10.0
7.5
5.0
1.0
1.5
2.0
2.5
Peso (em toneladas)
cambio
Automatico
Manual

Exemplo

```
qplot(peso,kpl,
      data = modelo,
      color = cambio,
      shape = cambio,
      geom = c("point","smooth"),
      xlab = "Peso (em toneladas)",
      ylab = "Kilometros por litro",
      main = "Modelo de Regressão")
```

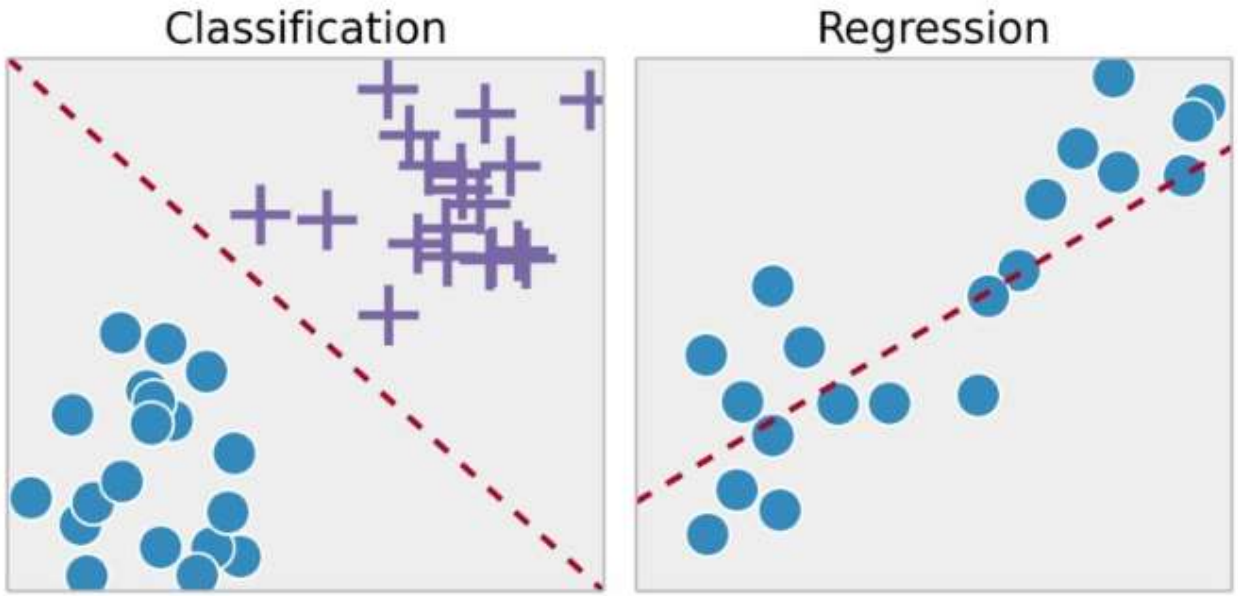
Digite o código acima e o *modelo* deve aparecer

Exemplo



Manual tende a ser melhor, além disso, dentro de cada categoria, há carros ótimos e ruins, eles são ...

Em uma imagem



Fonte: https://www.kdnuggets.com/

Hands On!

1. Use exemplo de *mtCars* para o arquivo ‘GeyserUFSM’. Crie um pequeno documento .pdf para descrever seus modelos. Este deve conter os coeficientes, os plots, e os resultados para os valores 200, 230, 245, e 270.

Obs: não há necessidade de analisar a saída.