Relatório Projeto e Aplicação de Mineração de Dados

Liz Alexandrita de Souza Barreto

02/12/2021

Universidade de Franca
RGM: 21125066

Preparação do Ambiente e Análise dos dados

Este relatório foi feito com o Rmarkdown!

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(farff)
library(rpart)
library(rpart.plot)
```

Leitura do arquivo e entendimento dos campos

```
iris_data <- readARFF('iris.arff')
```

```
## Parse with reader=readr : iris.arff
```

```
## header: 0.004000; preproc: 0.000000; data: 0.206000; postproc: 0.000000; total: 0.210000
```

```
#afour <- read_csv('A4.csv')
#stumat <- read_csv('student-mat.csv')
#ans <- read_tsv('4300Answers.tsv')
#twt <- read_tsv('TweetsNeutralNews.tsv')

glimpse(iris_data)
```

```
## Rows: 150
## Columns: 5
## $ sepallength <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4.8~
## $ sepalwidth  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.4~
## $ petallength <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6~
```

```
## $ petalwidth  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.2~
## $ class       <fct> Iris-setosa, Iris-setosa, Iris-setosa, Iris-setosa, Iris-s~
```

```
#glimpse(afour)
#glimpse(ans)
#glimpse(stumat)
#glimpse(twt)
```

Introdução

Vou realizar dois métodos, árvore de decisão e regressão linear para tentar discriminar entre classes do dataset iris. O tratamento básico será tentar retirar os registros faltantes e escolha de atributos para a regressão linear.

Escolha e Qualificação dos dados do dataset escolhido

```
iris_data %>% summary
```

```
##   sepallength      sepalwidth     petallength      petalwidth
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.054   Mean   :3.759   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##              class
##  Iris-setosa    :50
##  Iris-versicolor:50
##  Iris-virginica :50
##
##
##
```

Matriz de covariância

A diagonal principal marca a variância dos atributos e os demais elementos é a covariância.

```
var(iris_data)
```

```
## Warning in var(iris_data): NAs introduced by coercion
```

```
##              sepallength  sepalwidth petallength petalwidth class
## sepallength   0.68569351 -0.03926846   1.2736823  0.5169038    NA
## sepalwidth   -0.03926846  0.18800403  -0.3217128 -0.1179812    NA
## petallength   1.27368233 -0.32171275   3.1131794  1.2963875    NA
## petalwidth    0.51690380 -0.11798121   1.2963875  0.5824143    NA
## class                NA          NA          NA         NA    NA
```

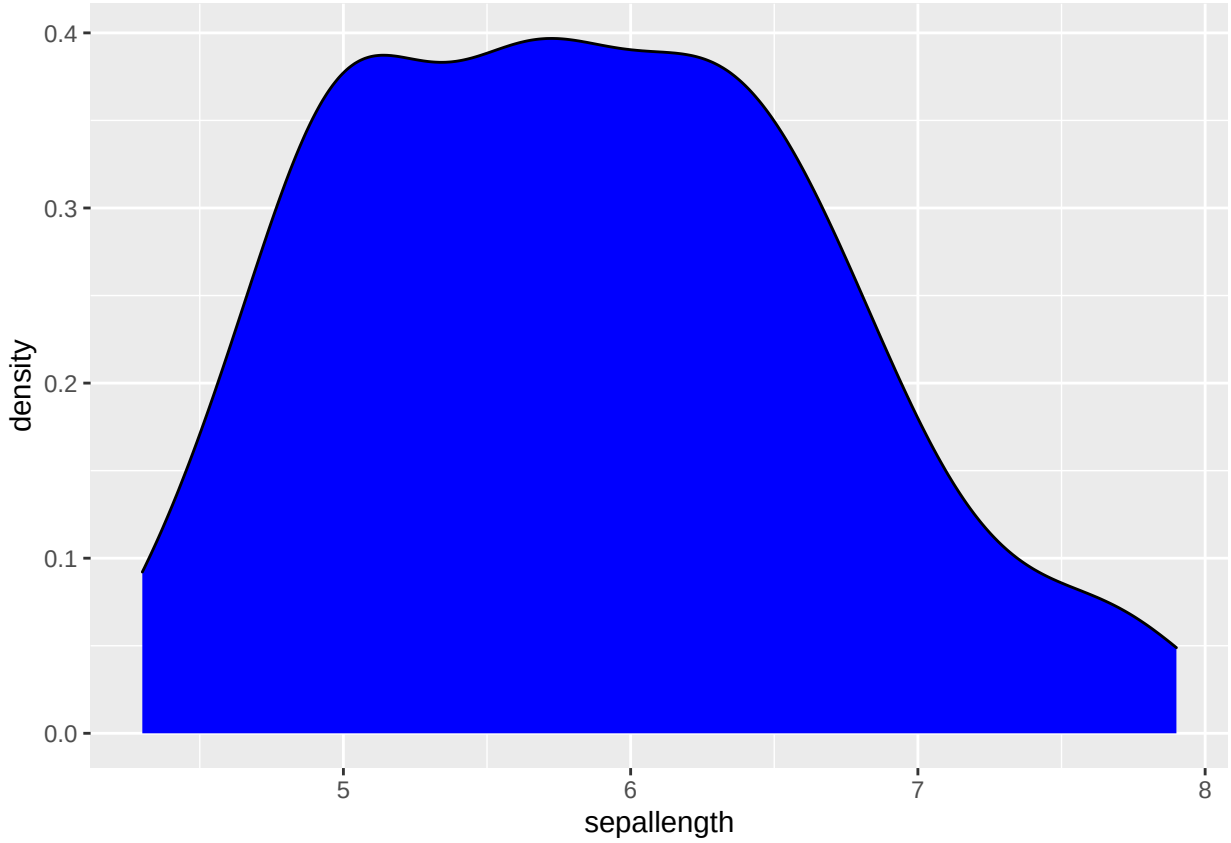
Verificação e tratamentos do dataset

```
nas <- colSums(is.na.data.frame(iris_data))
nas
```

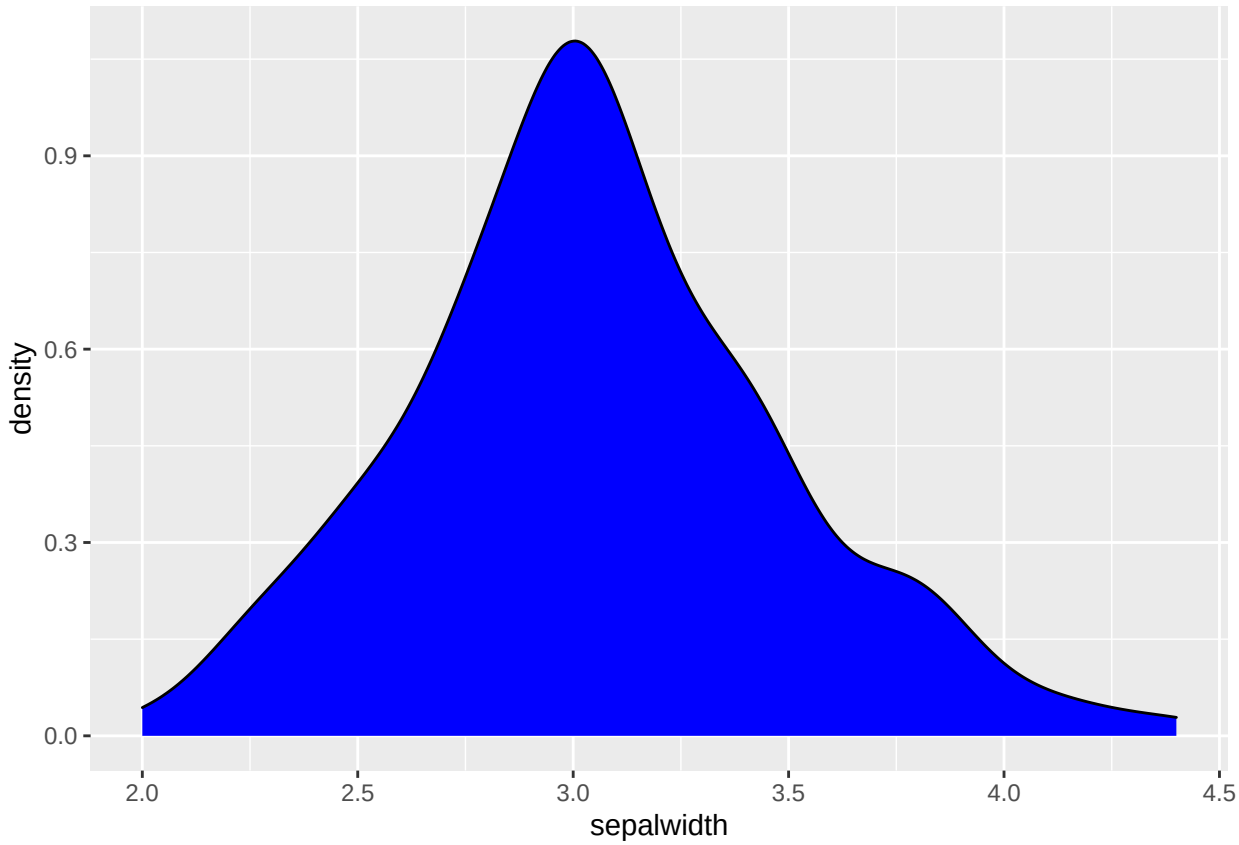
```
## sepallength  sepalwidth petallength  petalwidth       class
##           0           0           0           0           0
```

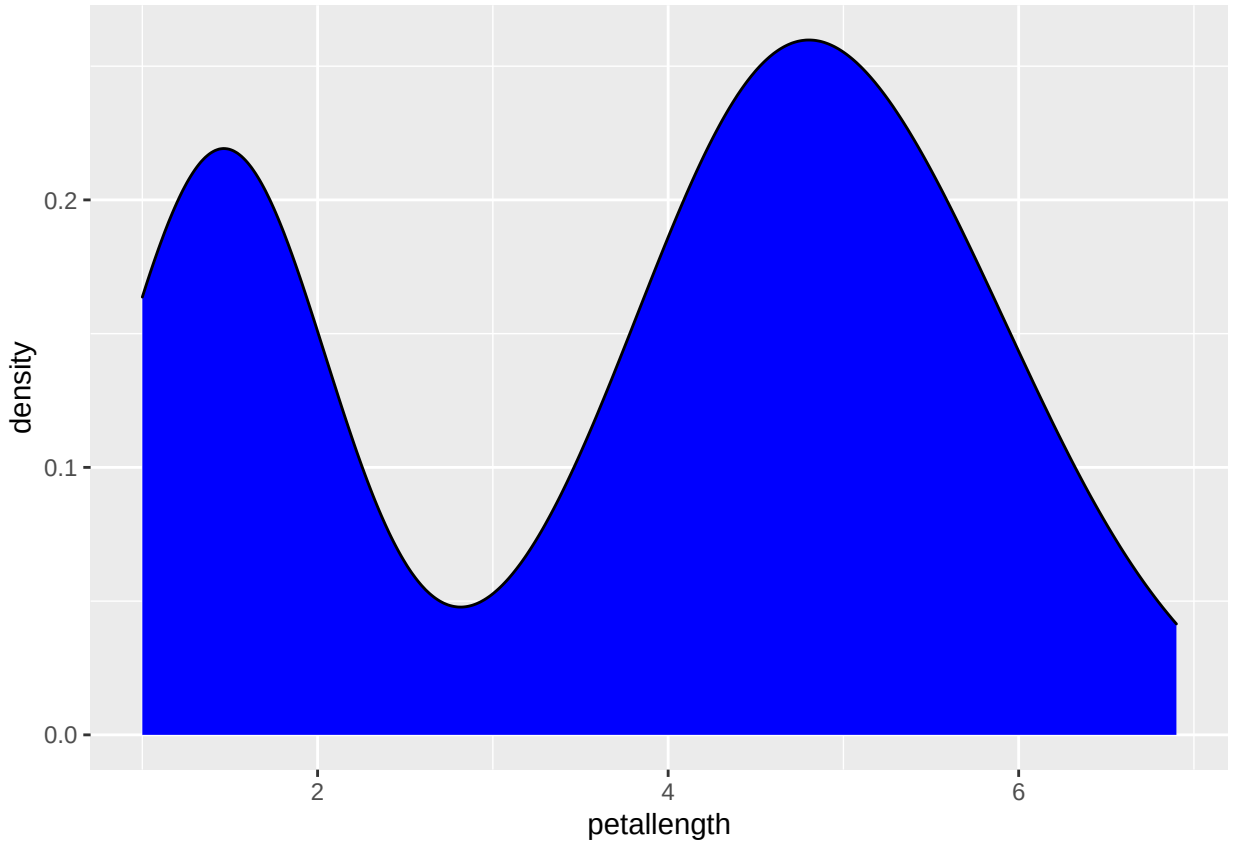
Gráfico de Densidade dos Atributos

```
iris_data %>% ggplot(aes(`sepallength`)) +
  geom_density(fill = "blue")
```

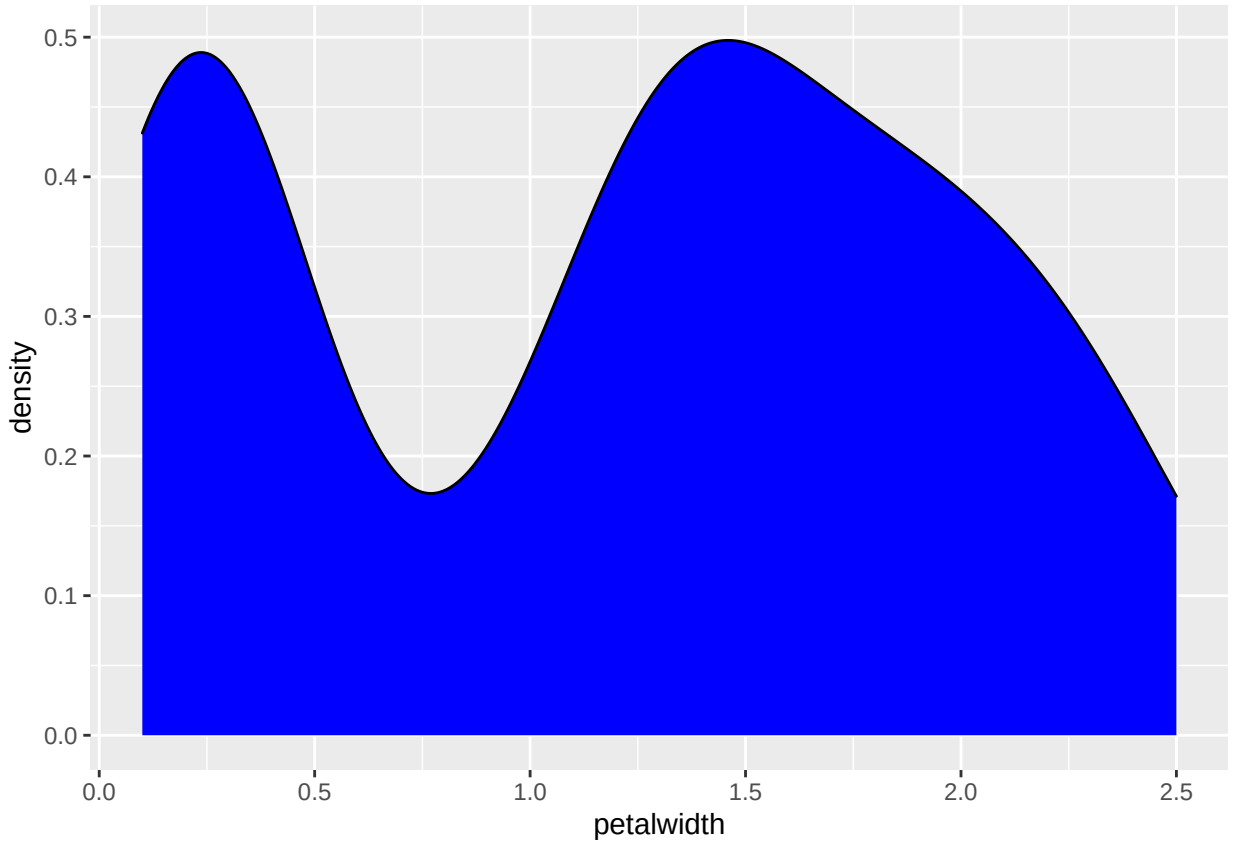



```
iris_data %>% ggplot(aes(`sepalwidth`)) +
  geom_density(fill = "blue")
```


density
0.9
0.6
0.3
0.0
2.0
2.5
3.0
3.5
4.0
4.5
sepalwidth


density
0.2
0.1
0.0
2
4
6
petallength

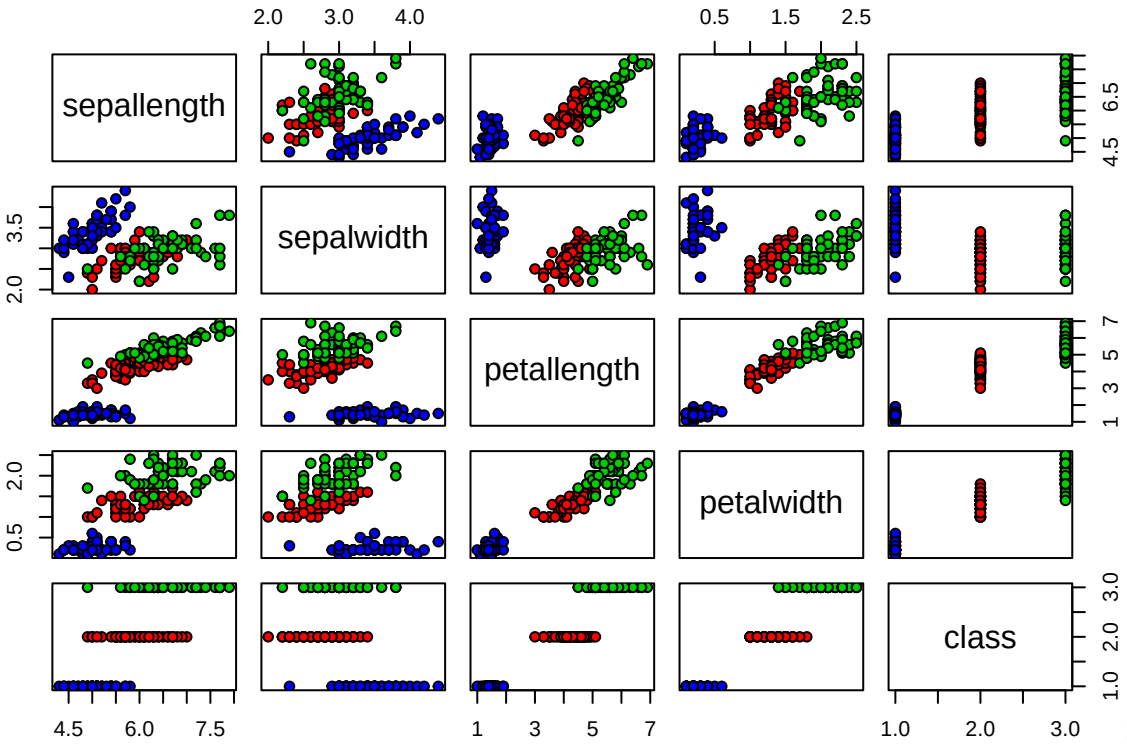
```r
iris_data %>% ggplot(aes(`petalwidth`)) +
  geom_density(fill = "blue") # esse tbm - agrupamento
```



Estudo gráfico

Quero analisar quais 2 atributos são interessantes para se criar uma árvore de decisão e uma regressão linear.

```r
iris_data %>% plot(pch = 21, bg=c("blue", "red", "green3")[unclass(iris_data$class)])
```



Decidi usar sepalwidth e petalwidth para a regresão linear.

Árvore de decisão

O modelo usa a classe e todos os atributos.

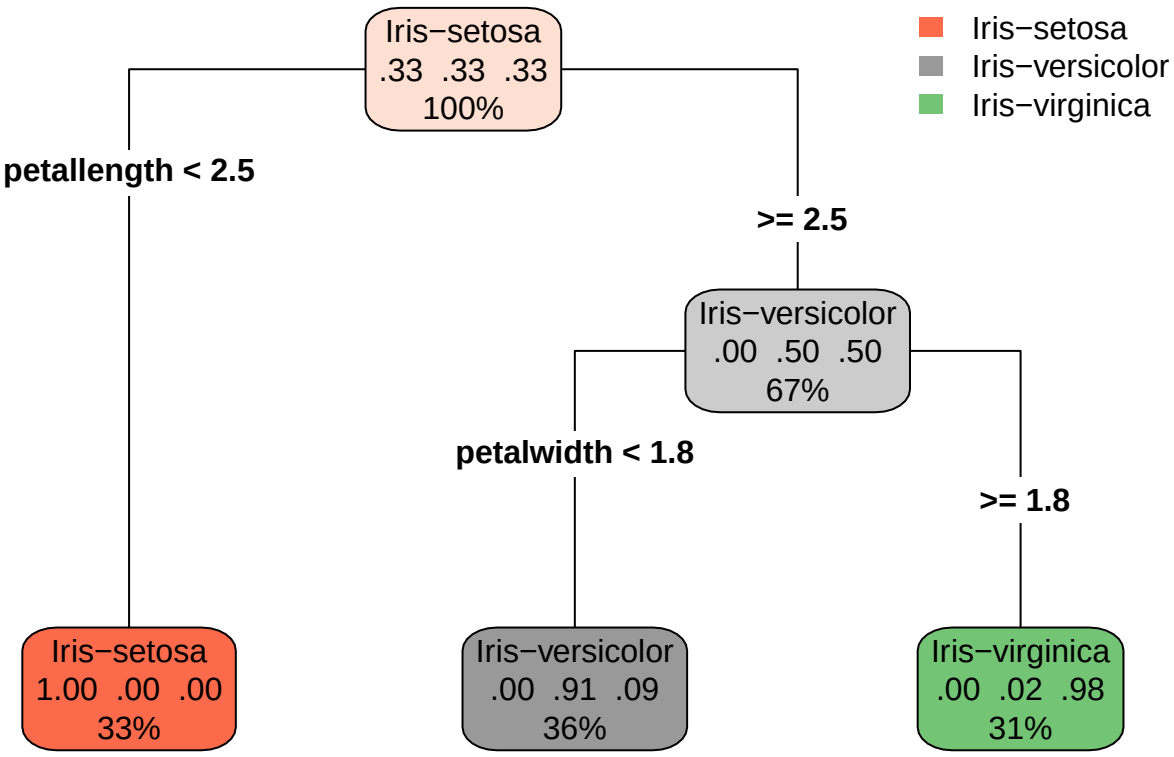
```
# modelinho <- tree(class ~ sepallength + sepalwidth + petallength + petalwidth, iris_data)
#modelinho <- rpart(class ~ petallength + petalwidth, iris_data, method = "class", x = TRUE)
modelinho <- rpart(class ~., iris_data, method = "class")

modelinho
```

```
## n= 150
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
## 1) root 150 100 Iris-setosa (0.33333333 0.33333333 0.33333333)
##   2) petallength< 2.45 50   0 Iris-setosa (1.00000000 0.00000000 0.00000000) *
##   3) petallength>=2.45 100  50 Iris-versicolor (0.00000000 0.50000000 0.50000000)
##     6) petalwidth< 1.75 54   5 Iris-versicolor (0.00000000 0.90740741 0.09259259) *
##     7) petalwidth>=1.75 46   1 Iris-virginica (0.00000000 0.02173913 0.97826087) *
```

A cada passo ele aumenta o peso para decidir se altera a classe do registro. Note que ele escolheu sozinho 2 atributos que melhor caracterizam a decisão de classificar esse dataset.

```
rpart.plot(modelinho, type = 4)
```

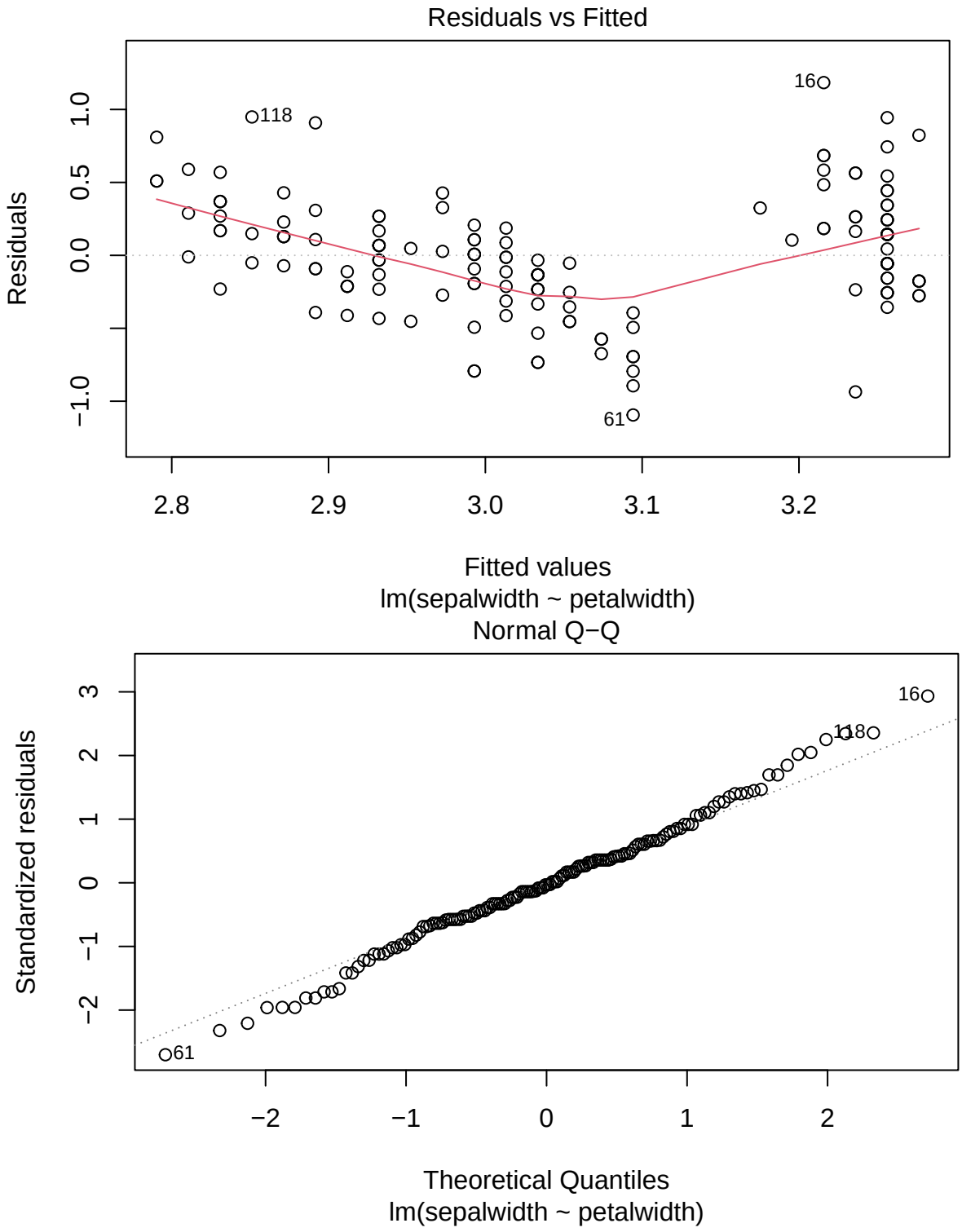


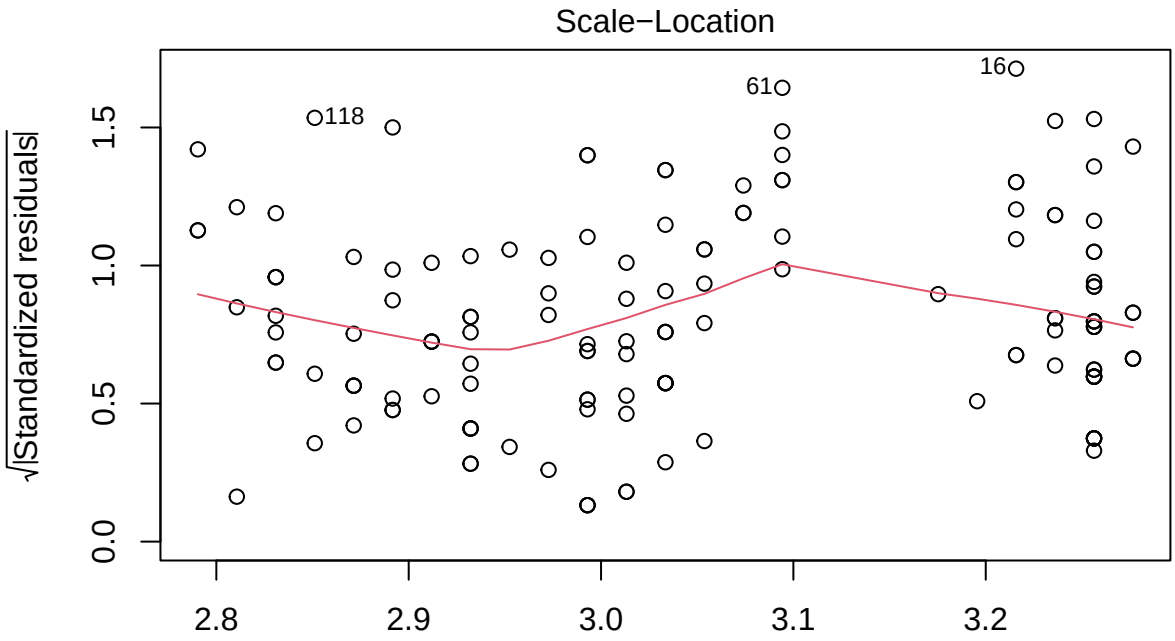
Regressão Linear com os atributos escolhidos

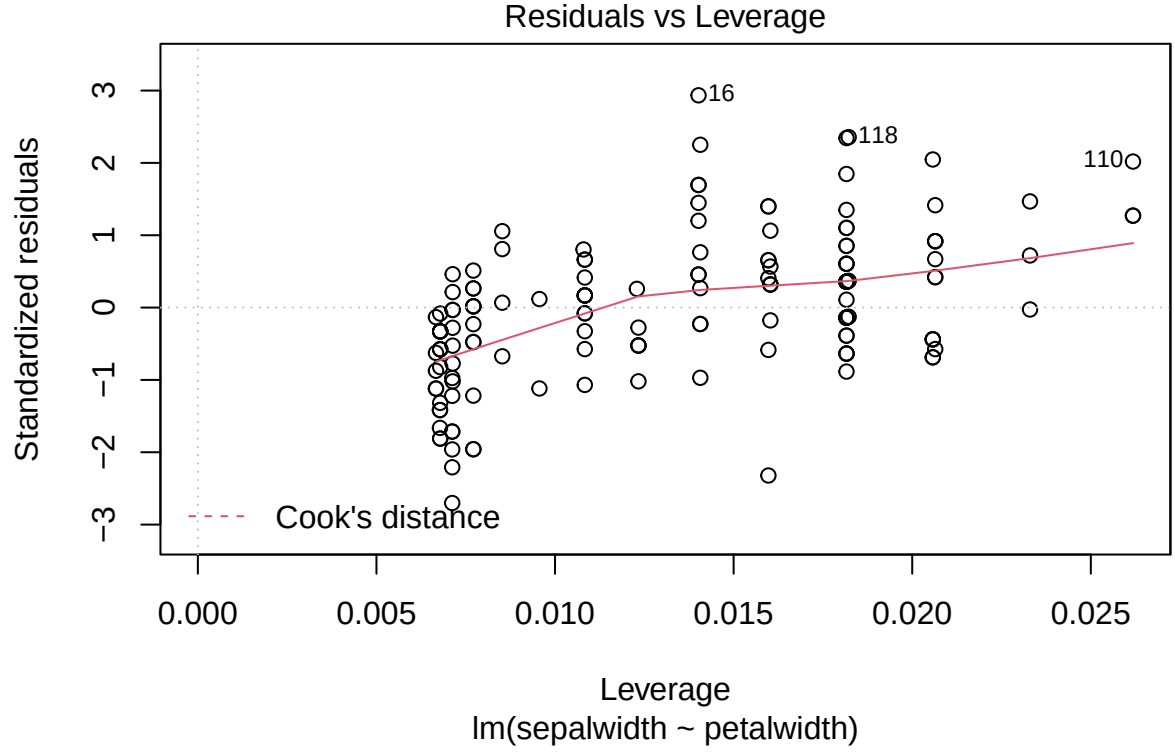
```
modelinho2 <- lm(sepalwidth ~ petalwidth, iris_data)
modelinho2
```

```
##
## Call:
## lm(formula = sepalwidth ~ petalwidth, data = iris_data)
##
## Coefficients:
## (Intercept)   petalwidth
##      3.2968      -0.2026
```

```
modelinho2 %>% plot()
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
−1.0
118
16
61
2.8
2.9
3.0
3.1
3.2
Fitted values
lm(sepalwidth ~ petalwidth)
Normal Q−Q
Standardized residuals
3
2
1
0
−1
−2
16
118
61
−2
−1
0
1
2
Theoretical Quantiles
lm(sepalwidth ~ petalwidth)



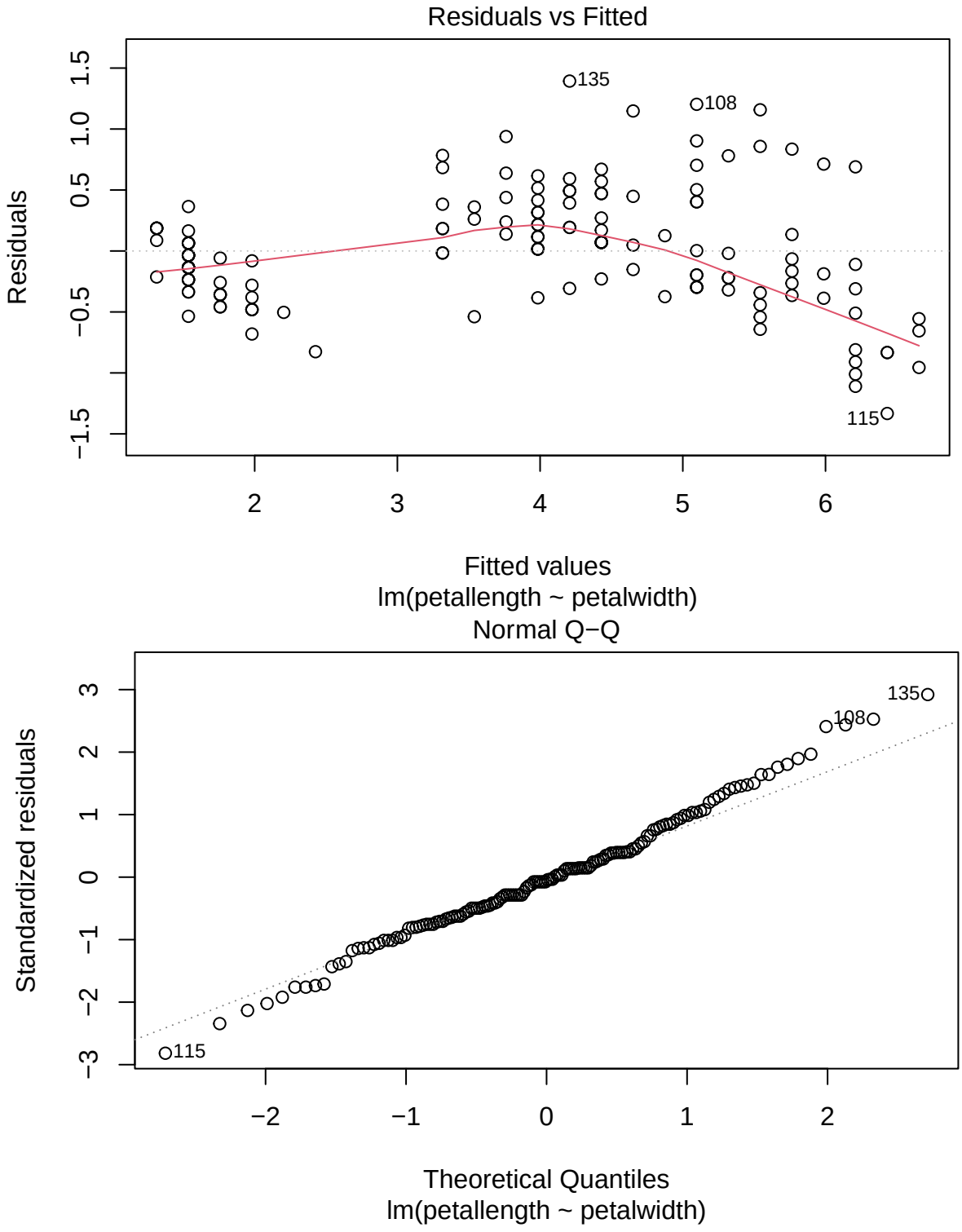


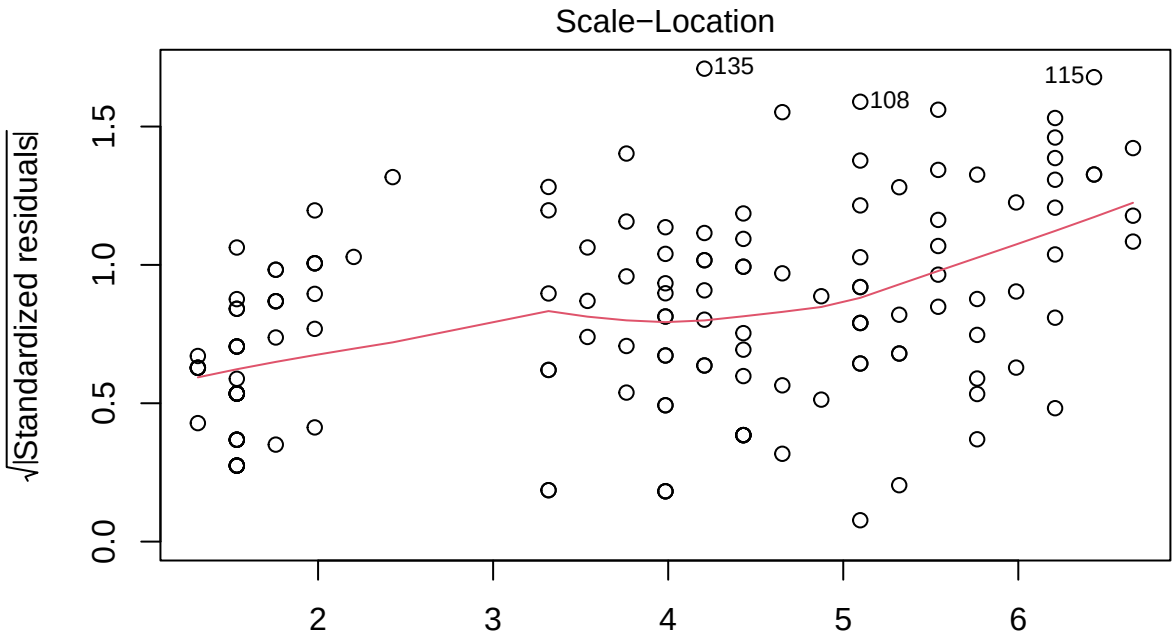

```
modelinho3 <- lm(petallength ~ petalwidth, iris_data)
modelinho3
```

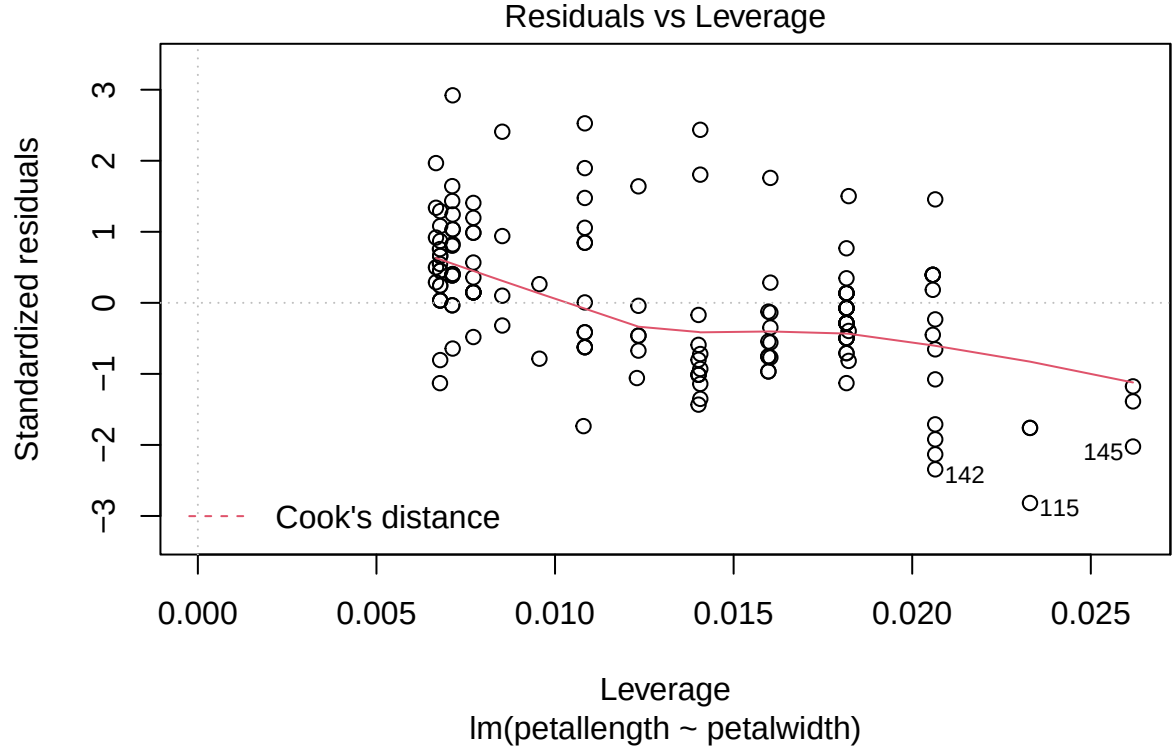
```
##
## Call:
## lm(formula = petallength ~ petalwidth, data = iris_data)
##
```

```
## Coefficients:
## (Intercept)   petalwidth
##       1.091        2.226
```

```
modelinho3 %>% plot()
```

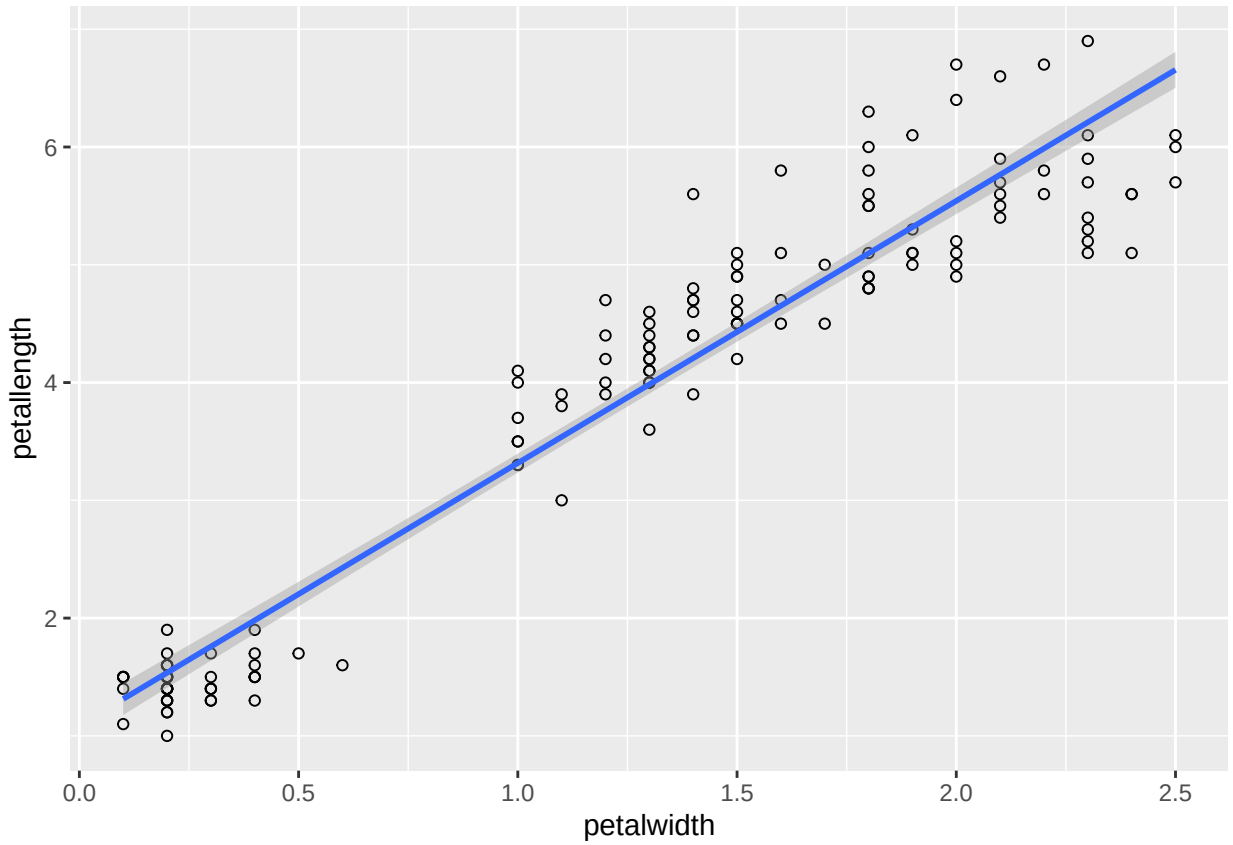






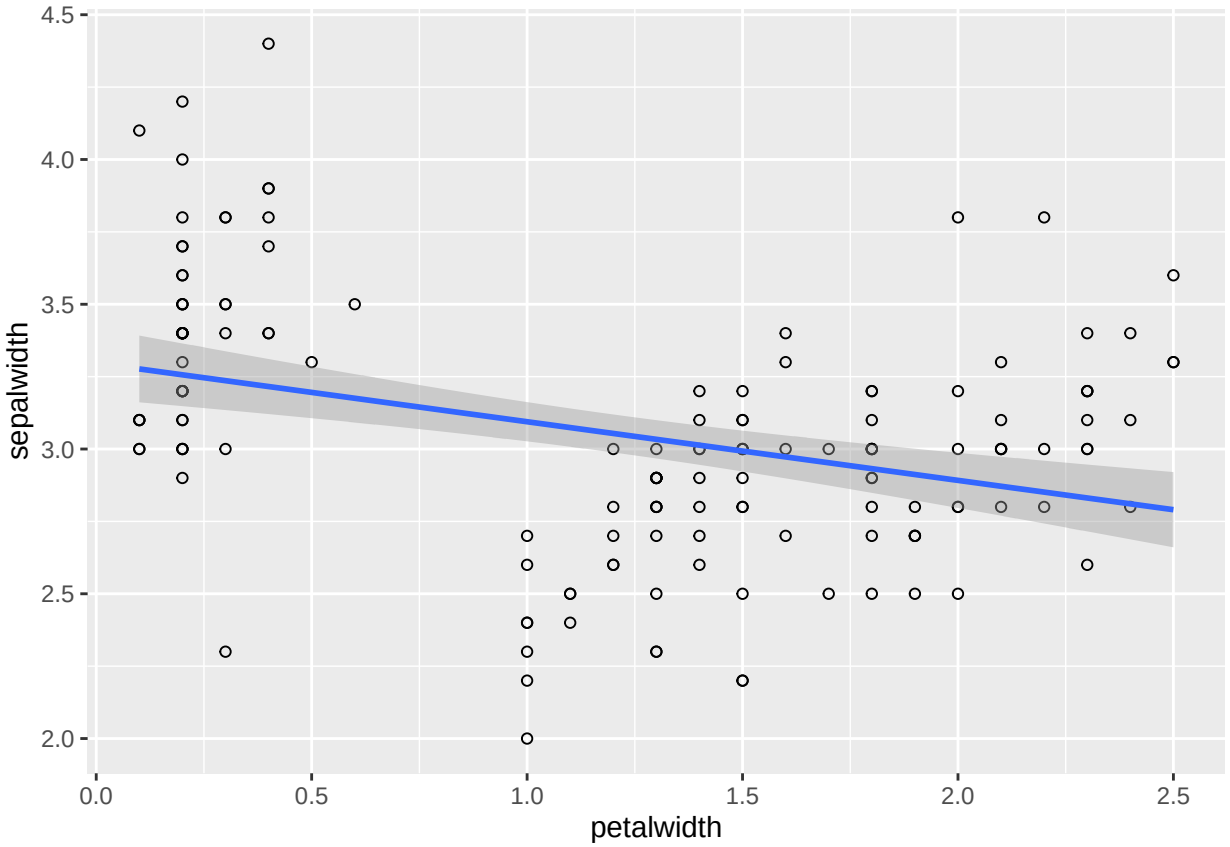

```
iris_data %>% ggplot(aes(x=`petalwidth`, y=`petallength`)) +
    geom_point(shape=1) +
    geom_smooth(method=lm)
```

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




```
iris_data %>% ggplot(aes(x=`petalwidth`, y=`sepalwidth`)) +
    geom_point(shape=1) +
    geom_smooth(method=lm)
```

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



A regressão nesse gráfico de dispersão mostra que é preciso de usar mais do que esses dois atributos para diferenciar nas classes rotuladas.

Conclusões

A classe setosa é facilmente identificável dentre as 3 classes. Observamos também pela regressão que há uma correlação negativa entre as variáveis escolhidas (largura de sepal e pétala) no contexto dos 3 grupos, no entanto é positiva em cada subgrupo. Já na regressão dos atributos escolhidos pelo algoritmo da árvore de decisão (as medidas da pétala) conseguimos realizar o recorte pelos valores em x e por isso a forte covariância entre essas medidas facilita a discriminação das classes mesmo olhando apenas para a reta de regressão.