Arvore de decisÃ£o no R: exemplo

Patricia Kuyven

02/10/2018

Exemplo de Árvore de Decisão com a base do Titanic

```
# Vamos ler a base do Titanic, busque o arquivo Titanic3.csv que você copiou do material da aula de 04/
# A variável "survived" indica se o passageiro sobreviveu (1) e se não sobreviveu (0).
library(readr)
titanic3 <- read_csv("~/GitHub/GeneralRepositoriesUnisinos/PosUnisinosIntroducaoPythonR/Aula06/titanic3
```

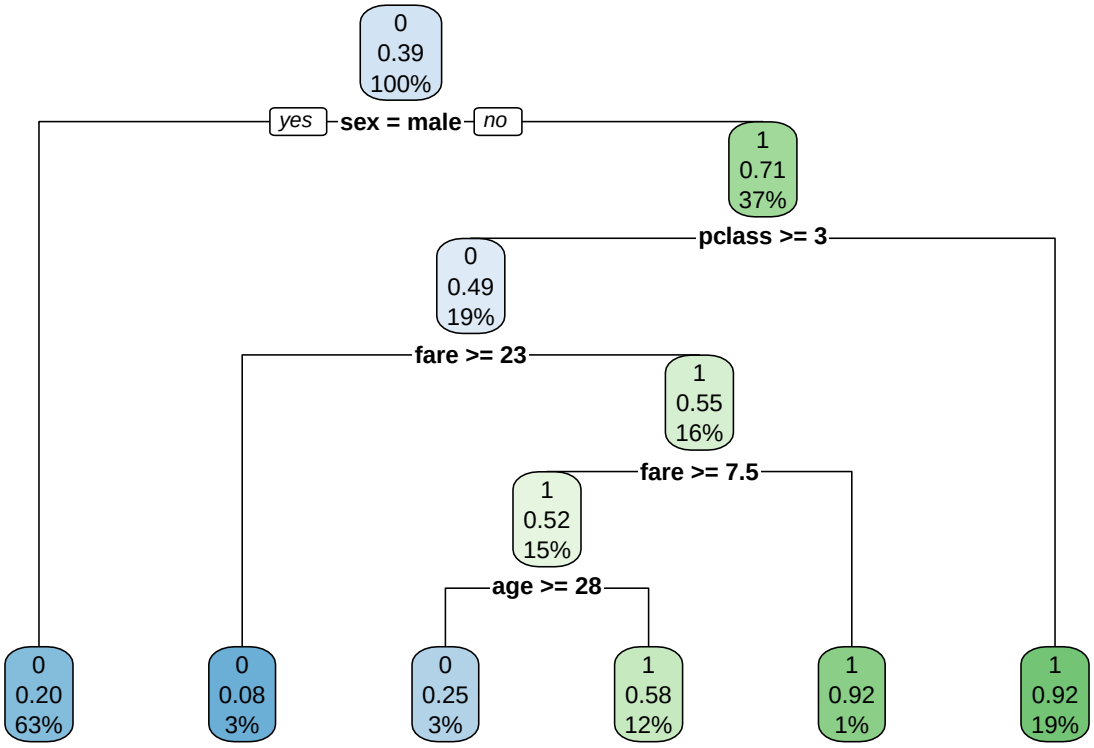
```
## Parsed with column specification:
## cols(
##   pclass = col_integer(),
##   survived = col_integer(),
##   name = col_character(),
##   sex = col_character(),
##   age = col_double(),
##   sibsp = col_integer(),
##   parch = col_integer(),
##   ticket = col_character(),
##   fare = col_double(),
##   cabin = col_character(),
##   embarked = col_character(),
##   boat = col_character(),
##   body = col_integer(),
##   home.dest = col_character()
## )
```

```
titanic3$survived <- as.factor(titanic3$survived)
```

```
# Vamos separar a base em treino e teste

## Gera indices da base treino e teste
set.seed(1234)
train_index = sample(1:nrow(titanic3), 0.7*nrow(titanic3), replace = FALSE);

## Gera base treino e teste
train = data.frame(); train = titanic3[train_index,]; test = data.frame(); test = titanic3[-train_index
```

```
# Neste exemplo a árvore para sobrevivência será modelada a partir das variáveis explicativas sex, age,

# No R usamos a função rpart para trabalhar com "árvores de decisão".
library(rpart)
arvore <- rpart(survived ~ sex + age + pclass + fare, data = train)
# Se fizermos o summary do modelo, temos uma descrição detalhada dos resultados obtidos para o modelo
# summary(arvore)
```

```
# podemos visualizar a árvore
library(rpart.plot)
rpart.plot(arvore)
```



Interpretação da árvore:

Observando o primeiro node, verificamos que 39% dentre todos os passageiros da base de teste sobreviveram. Ao considerar somente o sexo marculino, apenas 20% sobreviveu. E, ao considerar o sexo feminino, 71% sobreviveu (elas representam 37% da base total). Dentre as mulheres, se a classe da passageira era >= que 2,5, ou seja, 3ª classe, 49% sobreviveu; já as que viajaram de 1ª e 2ª classe, 92% sobreviveu. A interpretação pode continuar dessa forma recursivamente.

```
# Vamos calcular a probabilidade de sobreviver "prob" e a "categoria prevista" para cada caso da base d
probabilidades <- predict(arvore, newdata = test, type = "prob")
classes <- predict(arvore, newdata = test, type = 'class')
hr_model_arvore <- cbind(test, classes)
# Resumir os resultados
library(ddalpha)
```

```
## Loading required package: MASS
```

```
## Loading required package: class
```

```
## Loading required package: robustbase
## Loading required package: sfsmisc
## Loading required package: geometry
## Loading required package: magic
## Loading required package: abind
```

```
library("caret")
```

```
## Loading required package: lattice
## Loading required package: ggplot2
```

```
confusionMatrix <- confusionMatrix(hr_model_arvore$classes, hr_model_arvore$survived)
confusionMatrix
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 236  51
##          1  14  92
##
##                Accuracy : 0.8346
##                  95% CI : (0.7941, 0.87)
##     No Information Rate : 0.6361
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6218
##  Mcnemar's Test P-Value : 7.998e-06
##
##             Sensitivity : 0.9440
##             Specificity : 0.6434
##          Pos Pred Value : 0.8223
##          Neg Pred Value : 0.8679
##              Prevalence : 0.6361
##          Detection Rate : 0.6005
##    Detection Prevalence : 0.7303
##       Balanced Accuracy : 0.7937
##
##        'Positive' Class : 0
##
```

```
## Obter data.frame com dados previstos e dados real para comparação
dados_graf_p1 <- test[,c(2,4,5,9)]
dados_graf_p2 <- probabilidades
previsao <- classes
dados_grafico <- data.frame(dados_graf_p1, dados_graf_p2, previsao)
```

```
## Montar GRÁFICO com dados previstos e dados reais para comparação
library(ggplot2)
data("dados_grafico", package = "ggplot2")
```

```
## Warning in data("dados_grafico", package = "ggplot2"): data set
## 'dados_grafico' not found
```

```
graf_disp <- ggplot(dados_grafico, aes(x=age, y=X1)) +
  geom_point(aes(col=survived)) +
  labs(subtitle = "Titanic",
       y = "Probabilidade de sobreviver", x = "Idade")

plot(graf_disp)
```

```
## Warning: Removed 71 rows containing missing values (geom_point).
```