

# TarefaFinal\_Fifa

Balboni

22/06/2021

```
library(caret)

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

library(rpart)
library(rpart.plot)
library(parallel)
library(iterators)
library(foreach)
library(lattice)
library(ggplot2)
```

## Carregando dataset

```
entrada = read.csv('C://Users//Balboni//Desktop//Mestrado//Aulas//Mineração de dados//R//TarefaFinal//D
```

## Paralelismo

```
library(doParallel)

cl <- makePSOCKcluster(4)
registerDoParallel(cl)
```

## Excluindo variavel que n quero

```
entrada$X <- NULL

entrada$Value[entrada$Value == "Menor que 4700"] = "Entre 4700 e 5700"
entrada$Value[entrada$Value == "Entre 4700 e 5700"] = "Menor que 5700"
```

## Separação dos dados

```
p <- 0.75
set.seed(1)
inTraining <- sample.int(n = nrow(entrada),
                          size = floor(p * nrow(entrada)),
                          replace = FALSE)
```

```
treinoData <- entrada[inTraining, ]
testeData <- entrada[-inTraining, ]
```

## prob de cada classe

```
prop.table(table(entrada$Value))
```

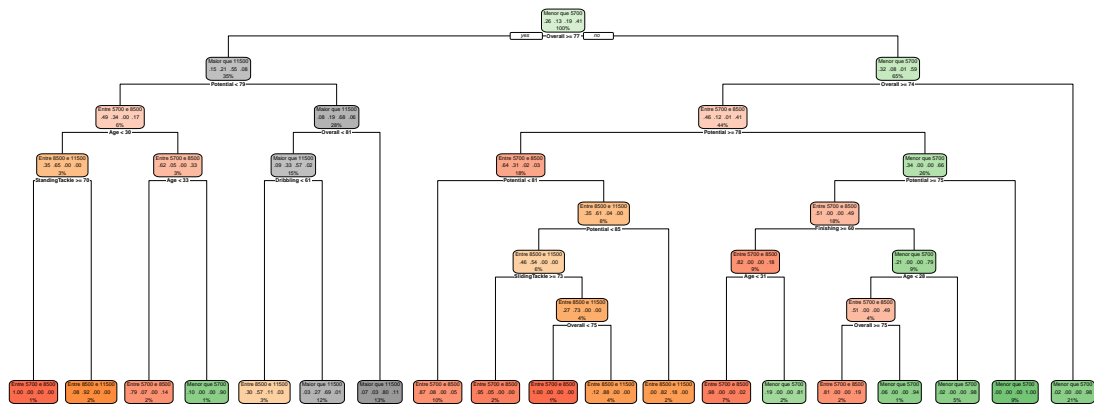
```
##
##  Entre 5700 e 8500 Entre 8500 e 11500  Maior que 11500  Menor que 5700
##           0.2749684           0.1245259           0.1997472           0.4007585
```

## Treinando

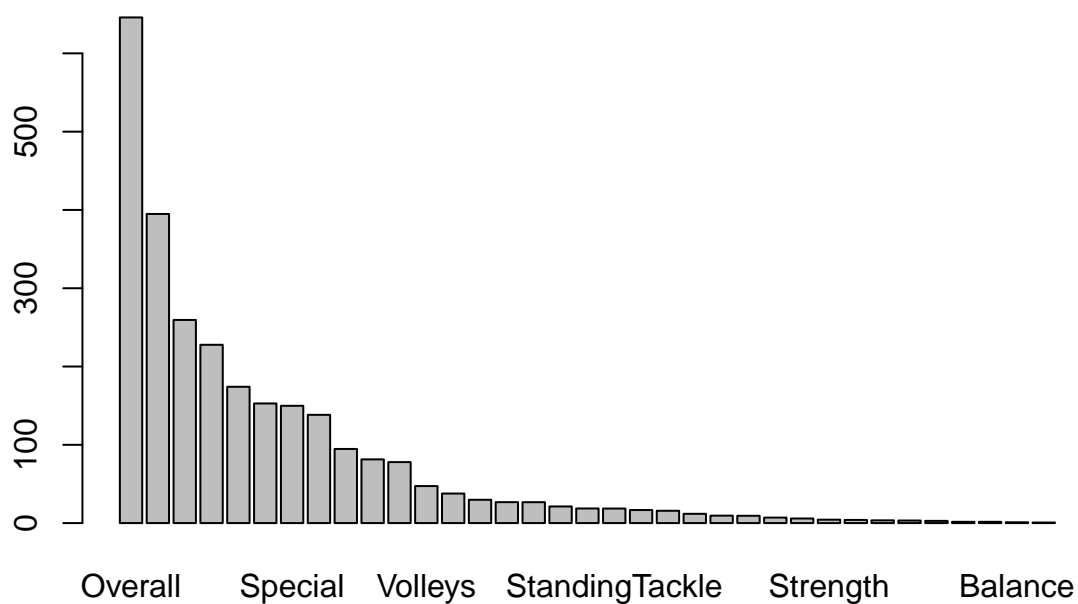
```
modelo1 = rpart(
  formula = treinoData$Value ~ .,
  data=treinoData,
  parms = list(split = "information"),
  cp = 0.002,
  control = rpart.control(
    sampling = "up",
    minsplit = 1,
    minbucket = 1,
    maxdepth = 30)
)
```

```
rpart.plot(modelo1)
```

■ Entre 5700 e 8500  
 ■ Entre 8500 e 11500  
 ■ Maior que 11500  
 ■ Menor que 5700



```
barplot(modelo1$variable.importance)
```



```

resultado1 = predict(modelo1, testeData, type = "class")
confusao1 = confusionMatrix(resultado1, as.factor(testeData$Value), mode = "prec_recall")
confusao1

```

```
## Confusion Matrix and Statistics
```

```
##
```

```
##           Reference
```

```
## Prediction      Entre 5700 e 8500 Entre 8500 e 11500 Maior que 11500
```

```
## Entre 5700 e 8500                100                9                0
```

```
## Entre 8500 e 11500                10               22                3
```

```
## Maior que 11500                   8               13               82
```

```
## Menor que 5700                    6                0                0
```

```
##           Reference
```

```
## Prediction      Menor que 5700
```

```
## Entre 5700 e 8500                  1
```

```
## Entre 8500 e 11500                  1
```

```
## Maior que 11500                      8
```

```
## Menor que 5700                     133
```

```
##
```

```
## Overall Statistics
```

```
##
```

```
##           Accuracy : 0.851
```

```
##           95% CI : (0.8121, 0.8846)
```

```
## No Information Rate : 0.3611
```

```
## P-Value [Acc > NIR] : < 2.2e-16
```

```
##
```

```
##           Kappa : 0.7919
```

```

##
## McNemar's Test P-Value : 0.0001529
##
## Statistics by Class:
##
##          Class: Entre 5700 e 8500 Class: Entre 8500 e 11500
## Precision          0.9091          0.61111
## Recall             0.8065          0.50000
## F1                 0.8547          0.55000
## Prevalence         0.3131          0.11111
## Detection Rate     0.2525          0.05556
## Detection Prevalence 0.2778          0.09091
## Balanced Accuracy  0.8848          0.73011
##
##          Class: Maior que 11500 Class: Menor que 5700
## Precision          0.7387          0.9568
## Recall             0.9647          0.9301
## F1                 0.8367          0.9433
## Prevalence         0.2146          0.3611
## Detection Rate     0.2071          0.3359
## Detection Prevalence 0.2803          0.3510
## Balanced Accuracy  0.9357          0.9532

```