Grupo AmaIA
Lander San Millán Langa
Amaia Acha Aristegui

Decision Trees for divorce prediction

1. JUSTIFICACIÓN DE LOS PARÁMETROS DE RPART()

En la fase del tuneo de hiperparametros hemos añadido los siguientes atributos en la función rpart():

- **Minsplit:** Mínimo número de observaciones que deben existir en un nodo antes de que el algoritmo intente realizar un split.

- **Minbucket:** Mínimo número de observaciones que tiene que haber en un nodo terminal

- **Maxdepth:** Máxima profundidad de cualquier nodo del árbol final.
  Root Node = 0

Hemos probado con diferentes valores de estos hipèrparametros y observado que al pasar valores altos a los atributos, se generaban árboles menos detallados. Finalmente hemos decidido quedarnos con valores bajos, así el árbol generado sea más detallado y nos permita clasificar las nuevas instancias de manera más restrictiva.

- **Minsplit = 3**
  Este valor es 3 para mantener la relación con minbucket.

- **Minbucket = 1**
  Según rdocumentation.org el valor adecuado de este atributo tiene que ser igual a Minsplit/3.

- **Maxdepth = 5**
  Al parecer, con los atributos anteriores no se generan árboles con profundidad superior a 3, por lo que este atributo se podría sustituir por dicho valor.

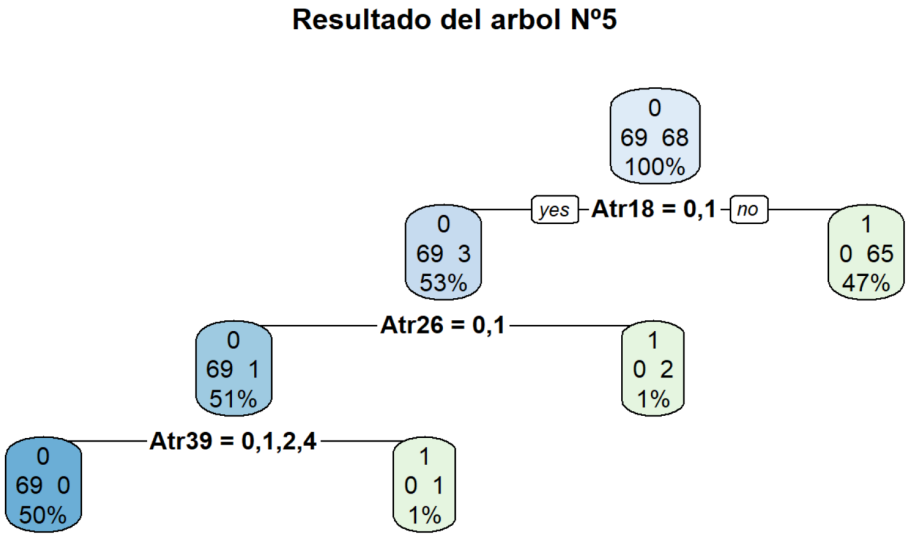
2. ATRIBUTOS MÁS RELEVANTES DE LOS MODELOS

En la siguiente tabla se muestra la precisión y los 5 atributos más relevantes de cada uno de los árboles de decisión generados:

Nº	Precisión del Árbol	5 atributos más relevantes (orden descendente)
1	0.90909091	20 - 17 - 18 - 19 - 11
2	0.93939394	17 - 14 - 19 - 21 - 9
3	0.96969697	18 - 16 - 19 - 20 - 40
4	0.93939394	18 - 19 - 16 - 17 - 20
5	1	18 - 16 - 20 - 9 - 17
6	0.96969697	18 - 16 - 20 - 30 - 40
7	0.93939394	11 - 16 - 19 - 20 - 40
8	0.93939394	11- 16 - 20 - 9 - 15
9	1	18 - 16 - 19 - 20 - 29
10	0.96969697	18 - 16 - 20 - 21 - 30

3. IMAGEN DEL MEJOR RESULTADO

A continuación, se muestra la imagen del árbol que mejor accuracy ha obtenido entre los 10 árboles previamente generados. Se muestra la imagen del 5.º árbol aunque el 9.º sería igual de bueno.



Acccuracy = 1

Decision Trees Divorce - AmaIA

Clean Environment & set path location

```
# Clear plots
if(!is.null(dev.list())) dev.off()
# Clear console
cat("\014")
# Clean workspace
rm(list=ls())
# Set working directory
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
```

Intall required packages

```
library(lattice)
library(ggplot2)
library(caret)
library(rpart)
library(rpart.plot)
```

Read data from CSV

```
filename = "../data/divorce.csv"
data <- read.csv(file = filename, sep =";", header = TRUE)
```

Convert columns to factors

```
index <- 1:ncol(data)
data[ , index] <- lapply(data[ , index], as.factor)
```

Set the Percentaje of training examples

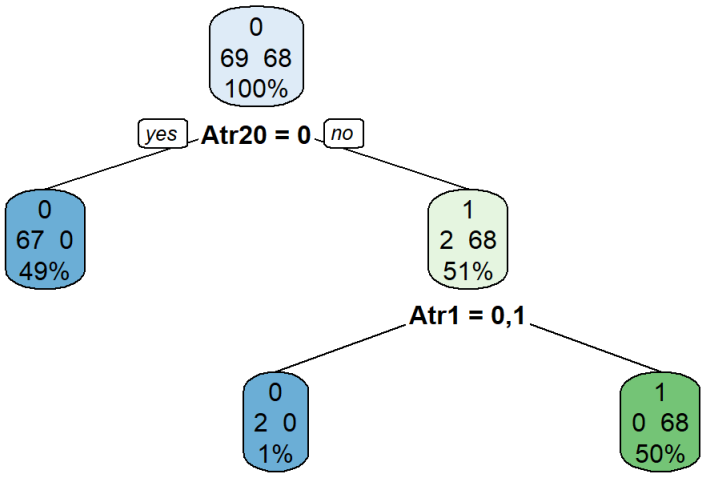
```
training_p <- 0.8
```

Generate 10 Decission Trees

```r
for (i in 1:10) {
  # Generate data partition 80% training / 20% test. The result is a vector with the indexes
  # of the examples that will be used for the training of the model.
  training_indexes <- createDataPartition(y = data$Class, p = training_p, list = FALSE)

  # Split training and test data
  training_data <- data[training_indexes, ]  # Extract training data using training_indexes
  test_data     <- data[-training_indexes, ] # Extract data with the indexes not included in training_indexe
s

  # Create Linear Model using training data. Formula = all the columns except Class
  model <- rpart(formula = Class ~., data = training_data, minsplit= 3 , minbucket=1, maxdepth=5)

  # Make the prediction using the model and test data
  prediction <- predict(model, test_data, type = "class")

  # Calculate accuracy using Confusion Matrix
  prediction_results <- table(test_data$Class, prediction)
  matrix <- confusionMatrix(prediction_results)
  accuracy <- matrix$overall[1]
  print(paste0("Importancia de las variables:"))
  print(model$variable.importance)
  attrs <- names(model$variable.importance)

  print(paste0("Accuracy = ", round(accuracy, digits = 8)), quote = FALSE)

  # Print the rules that represent the Tree
  rpart.rules(model, extra = 9, cover = TRUE, digits = 8)

  # Plot tree (this method is slow, wait until pot is completed)
  #renderPlot({
    rpart.plot(model,
               type = 2,
               extra = 101,
               fallen.leaves = FALSE,
               main = paste0("Resultado del arbol N°", as.character(i)),
               sub = paste0("Acccuracy = ", round(accuracy, digits = 8)))
  #})
}
```

```
## [1] "Importancia de las variables:"
##     Atr20     Atr17     Atr18     Atr19     Atr11      Atr9      Atr1
## 64.610636 60.753285 60.753285 60.753285 59.788947 59.788947  3.885714
## [1] Accuracy = 0.90909091
```

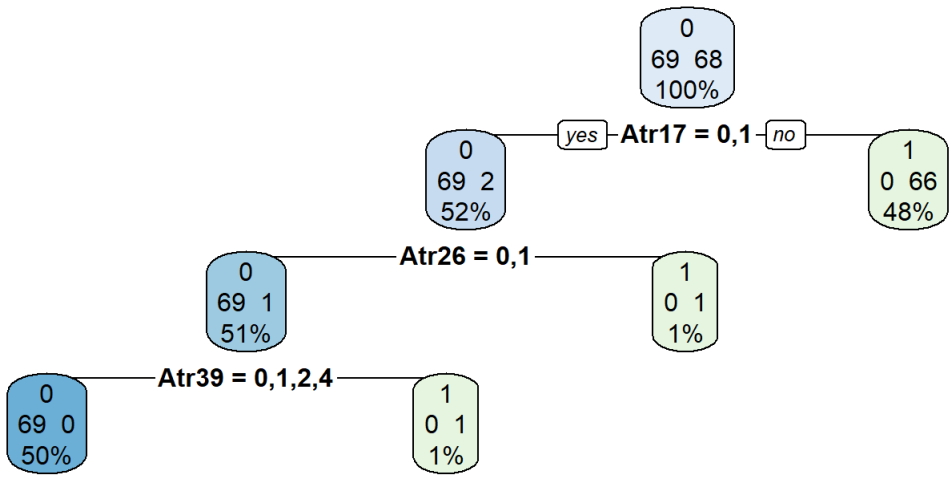
Resultado del arbol Nº1



Acccuracy = 0.90909091

```
## [1] "Importancia de las variables:"
##     Atr17     Atr14     Atr19     Atr21      Atr9     Atr15     Atr39     Atr26
## 64.609026 62.651177 62.651177 62.651177 62.651177 61.672252  1.971429  1.915895
## [1] Accuracy = 0.93939394
```

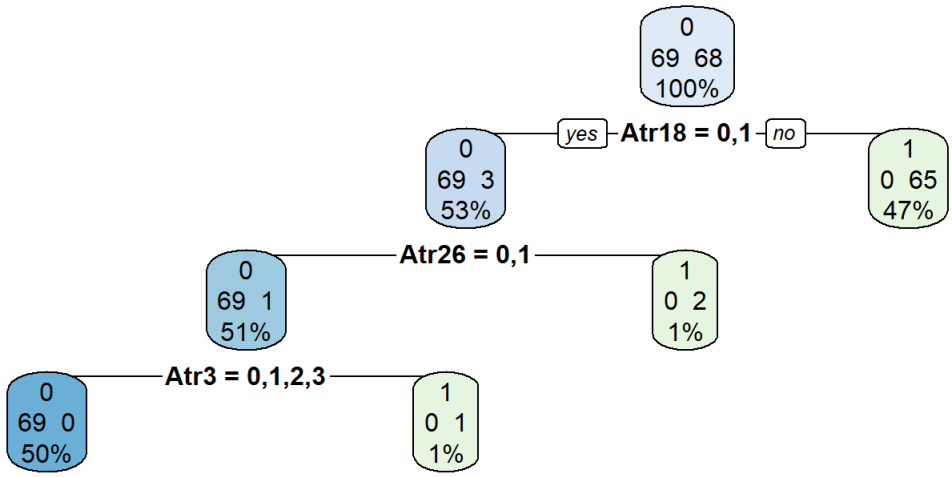
Resultado del arbol Nº2



Acccuracy = 0.93939394

```
## [1] "Importancia de las variables:"
##     Atr18     Atr16     Atr19     Atr20     Atr40      Atr9     Atr26      Atr3
## 62.746350 60.815693 60.815693 60.815693 60.815693 60.815693  3.778571  1.971429
## [1] Accuracy = 0.96969697
```

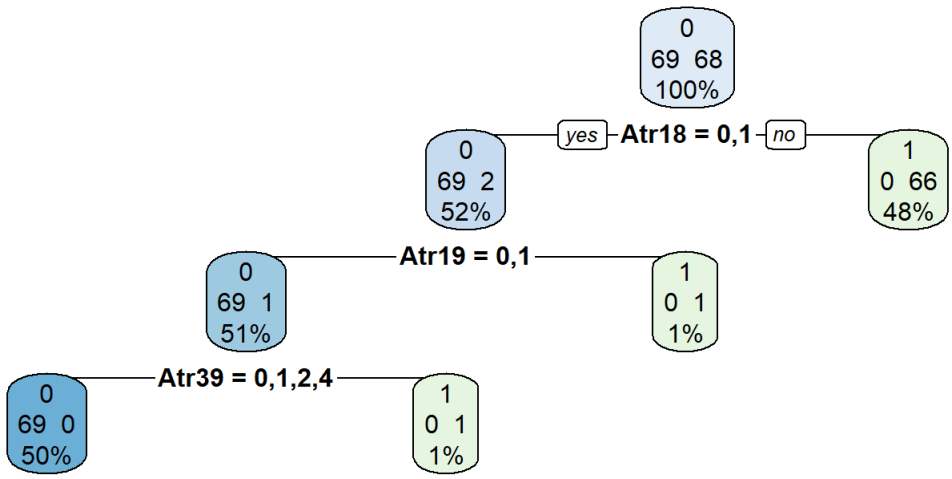
Resultado del arbol Nº3



Acccuracy = 0.96969697

```
## [1] "Importancia de las variables:"
##     Atr18     Atr19     Atr16     Atr17     Atr20      Atr9     Atr39
## 64.609026 64.567073 62.651177 62.651177 62.651177 62.651177  1.971429
## [1] Accuracy = 0.93939394
```

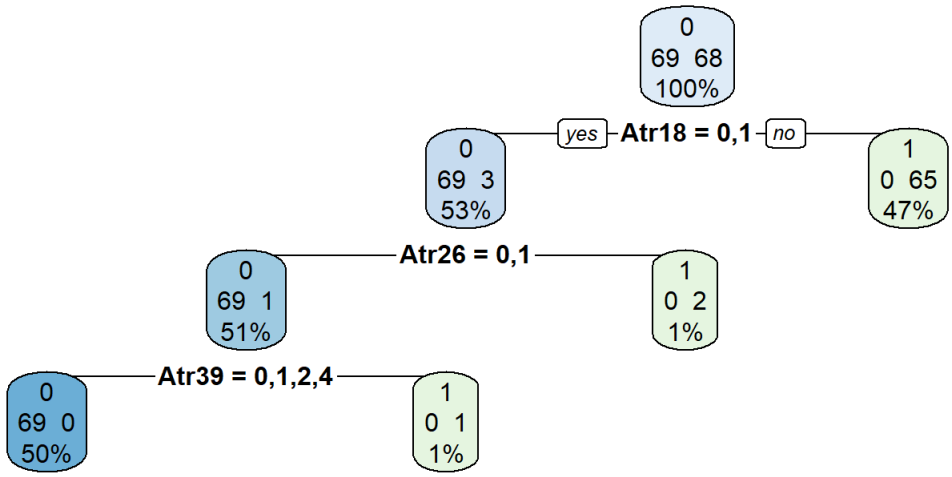
Resultado del arbol Nº4



Acccuracy = 0.93939394

```
## [1] "Importancia de las variables:"
##     Atr18     Atr16     Atr20      Atr9     Atr17     Atr19     Atr26     Atr39
## 62.746350 60.815693 60.815693 60.815693 59.850365 59.850365  3.778571  1.971429
## [1] Accuracy = 1
```

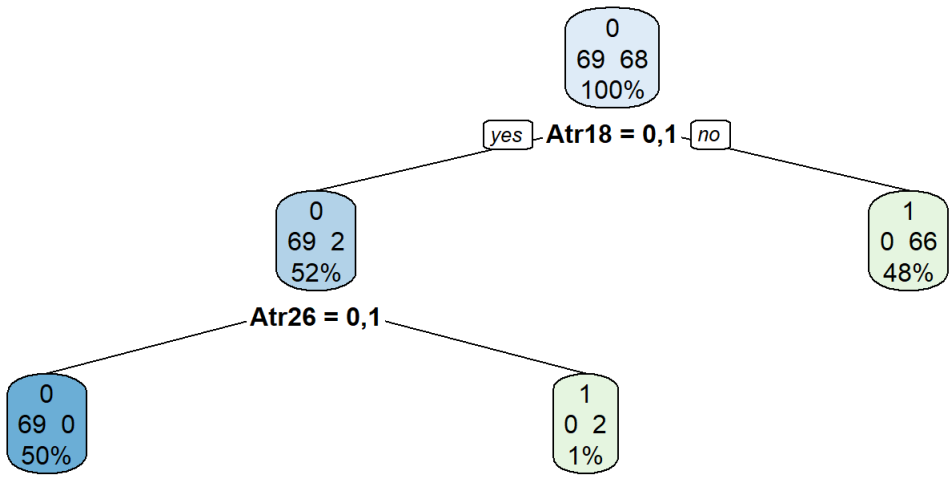
Resultado del arbol Nº5



Acccuracy = 1

```
## [1] "Importancia de las variables:"
##     Atr18     Atr16     Atr20     Atr30     Atr40      Atr9     Atr26
## 64.609026 62.651177 62.651177 62.651177 62.651177 62.651177  3.887324
## [1] Accuracy = 0.96969697
```

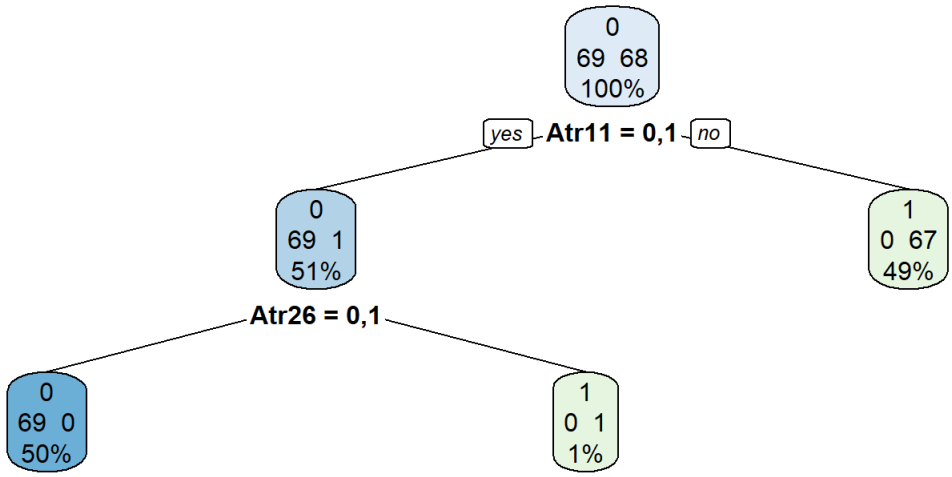
Resultado del arbol Nº6



Acccuracy = 0.96969697

```
## [1] "Importancia de las variables:"
##     Atr11     Atr16     Atr19     Atr20     Atr40      Atr9     Atr26
## 66.524922 64.539103 64.539103 64.539103 64.539103 64.539103  1.971429
## [1] Accuracy = 0.93939394
```

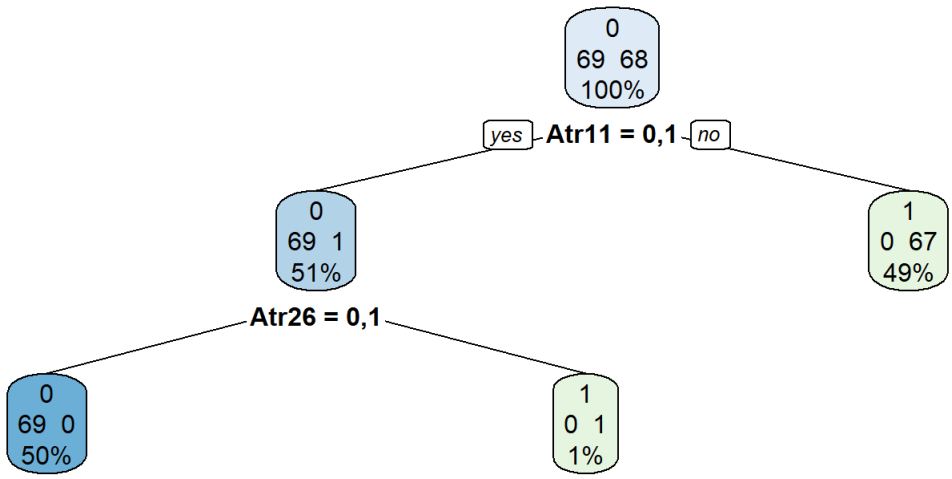
Resultado del arbol Nº7



Acccuracy = 0.93939394

```
## [1] "Importancia de las variables:"
##     Atr11     Atr16     Atr20     Atr40      Atr9     Atr15     Atr26
## 66.524922 64.539103 64.539103 64.539103 64.539103 63.546194  1.971429
## [1] Accuracy = 0.93939394
```

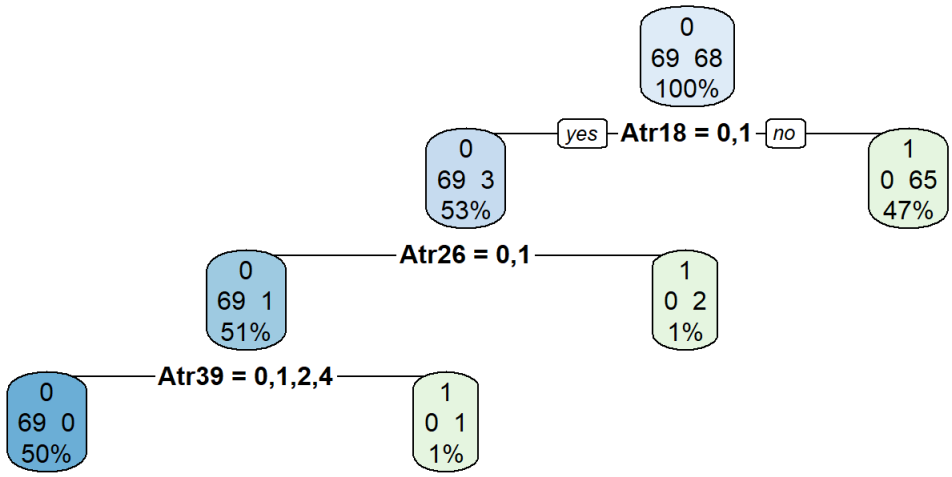
Resultado del arbol Nº8



Acccuracy = 0.93939394

```
## [1] "Importancia de las variables:"
##     Atr18     Atr16     Atr19     Atr20     Atr29      Atr9     Atr26     Atr39
## 62.746350 60.815693 60.815693 60.815693 60.815693 60.815693  3.778571  1.971429
## [1] Accuracy = 1
```

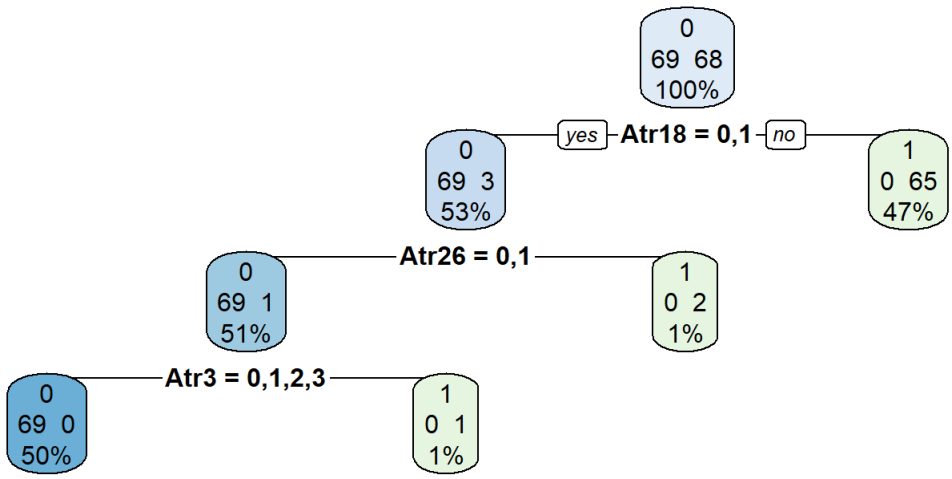
Resultado del arbol Nº9



Acccuracy = 1

```
## [1] "Importancia de las variables:"
##     Atr18     Atr16     Atr20     Atr21     Atr30      Atr9     Atr26      Atr3
## 62.746350 60.815693 60.815693 60.815693 60.815693 60.815693  3.778571  1.971429
## [1] Accuracy = 0.96969697
```

Resultado del arbol Nº10



Acccuracy = 0.96969697