Comparar modelos de Clasificación Logístico vs Arbol de Clasificación

Rubén Pizarro Gurrola

26/3/2022

# Objetivo

Comparar y evaluar modelos de clasificación lógistico y árbol de clasificación con datos y variables de la felicidad de personas.

# Descripción

La comparación se utilizan métricas a partir de la matriz de confusión. Se necesita la construcción de ambos modelos.

# Librerías

#install.packages("readr")  
#install.packages("caret")  
#install.packages("rpart.plot")  
  
library(readr) # Leer datos  
library(caret) # Partir datos y matriz de confusión  
library(ggplot2) # Para graficar  
library(dplyr) # filtros select mutate summarize, group\_by  
library(knitr) # Tabular data.frames amigable no tan textual  
library(rpart) # árbol  
library(rpart.plot) # vis árbol

# Datos

datos <- read.csv("https://raw.githubusercontent.com/rpizarrog/Machine-Learning-con-R/main/datos/estado%20de%20felicidad%20variables.csv")

## Explorar datos

str(datos)

## 'data.frame': 52 obs. of 10 variables:  
## $ genero : chr "MASCULINO" "FEMENINO" "MASCULINO" "FEMENINO" ...  
## $ esto.civil : chr "SOLTERO" "CASADO" "DIVORCIADO" "VIUDO" ...  
## $ edad : int 25 35 45 54 52 28 56 32 35 29 ...  
## $ satisfaccion.laboral : num 80 50 70 50 40 50 71.5 60 70 80 ...  
## $ satisfaccion.profesional: num 90 80 78 80 50 60 60.8 80 60 80 ...  
## $ vida.familiar : num 70 60 80 60 60 54 86.9 30 72 90 ...  
## $ vida.social : num 80 70 40 80 70 60 70.6 50 60 60 ...  
## $ salud : chr "BUENO" "MALO" "REGULAR" "BUENO" ...  
## $ dinero : num 90 30 70 20 60 50 70 20 70 80 ...  
## $ estado : chr "FELIZ" "NO FELIZ" "FELIZ" "NO FELIZ" ...

summary(datos)

## genero esto.civil edad satisfaccion.laboral  
## Length:52 Length:52 Min. :25.00 Min. : 24.00   
## Class :character Class :character 1st Qu.:34.00 1st Qu.: 50.00   
## Mode :character Mode :character Median :39.50 Median : 60.00   
## Mean :40.79 Mean : 64.03   
## 3rd Qu.:45.00 3rd Qu.: 80.00   
## Max. :88.00 Max. :100.00   
## satisfaccion.profesional vida.familiar vida.social salud   
## Min. : 40.00 Min. : 20.00 Min. : 20.00 Length:52   
## 1st Qu.: 60.00 1st Qu.: 50.00 1st Qu.: 50.00 Class :character   
## Median : 70.00 Median : 61.00 Median : 60.00 Mode :character   
## Mean : 68.98 Mean : 65.48 Mean : 64.57   
## 3rd Qu.: 80.00 3rd Qu.: 80.00 3rd Qu.: 80.00   
## Max. :100.00 Max. :100.00 Max. :100.00   
## dinero estado   
## Min. : 20.00 Length:52   
## 1st Qu.: 48.00 Class :character   
## Median : 60.00 Mode :character   
## Mean : 60.86   
## 3rd Qu.: 80.00   
## Max. :100.00

## head()

kable(head(datos, 10), caption = "Primeros 10")

Primeros 10

| genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MASCULINO | SOLTERO | 25 | 80.0 | 90.0 | 70.0 | 80.0 | BUENO | 90 | FELIZ |
| FEMENINO | CASADO | 35 | 50.0 | 80.0 | 60.0 | 70.0 | MALO | 30 | NO FELIZ |
| MASCULINO | DIVORCIADO | 45 | 70.0 | 78.0 | 80.0 | 40.0 | REGULAR | 70 | FELIZ |
| FEMENINO | VIUDO | 54 | 50.0 | 80.0 | 60.0 | 80.0 | BUENO | 20 | NO FELIZ |
| MASCULINO | CASADO | 52 | 40.0 | 50.0 | 60.0 | 70.0 | BUENO | 60 | NO FELIZ |
| FEMENINO | SOLTERO | 28 | 50.0 | 60.0 | 54.0 | 60.0 | MALO | 50 | NO FELIZ |
| MASCULINO | VIUDO | 56 | 71.5 | 60.8 | 86.9 | 70.6 | MALO | 70 | FELIZ |
| FEMENINO | DIVORCIADO | 32 | 60.0 | 80.0 | 30.0 | 50.0 | REGULAR | 20 | NO FELIZ |
| MASCULINO | CASADO | 35 | 70.0 | 60.0 | 72.0 | 60.0 | BUENO | 70 | FELIZ |
| FEMENINO | SOLTERO | 29 | 80.0 | 80.0 | 90.0 | 60.0 | MALO | 80 | FELIZ |

## tail()

kable(tail(datos, 10), caption = "Ultimos 10")

Ultimos 10

|  | genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 43 | FEMENINO | SOLTERO | 45 | 90 | 95 | 90 | 100 | BUENO | 90 | FELIZ |
| 44 | MASCULINO | VIUDO | 88 | 65 | 66 | 89 | 87 | REGULAR | 45 | NO FELIZ |
| 45 | MASCULINO | CASADO | 54 | 90 | 90 | 90 | 100 | BUENO | 75 | FELIZ |
| 46 | MASCULINO | SOLTERO | 80 | 80 | 80 | 90 | 90 | BUENO | 90 | FELIZ |
| 47 | MASCULINO | SOLTERO | 39 | 50 | 80 | 90 | 80 | REGULAR | 90 | FELIZ |
| 48 | FEMENINO | SOLTERO | 40 | 30 | 60 | 80 | 50 | REGULAR | 50 | NO FELIZ |
| 49 | FEMENINO | SOLTERO | 49 | 80 | 80 | 80 | 80 | BUENO | 70 | FELIZ |
| 50 | MASCULINO | SOLTERO | 39 | 60 | 80 | 48 | 50 | REGULAR | 50 | NO FELIZ |
| 51 | FEMENINO | SOLTERO | 30 | 100 | 100 | 100 | 100 | BUENO | 100 | NO FELIZ |
| 52 | FEMENINO | SOLTERO | 32 | 70 | 80 | 80 | 80 | BUENO | 80 | FELIZ |

# Transformar/Limpiar datos

Como parte de preparar los datos habrá que transformar la variabe estado: \* FELIZ = 1  
\* NO FELIZ = 0

datos <- datos %>%  
 mutate(estado.01 = if\_else(estado == "FELIZ", 1, 0))

# Particionar los datos

Partir los datos con una semilla de 2022 con el 70% para dtos de entrenamiento y el 30% para datos de validación.

## Sembrar semilla

set.seed(2022)

## Datos de entrenamiento

n <- nrow(datos)  
  
entrena <- createDataPartition(y = datos$estado, p = 0.70, list = FALSE, times = 1)  
  
entrena

## Resample1  
## [1,] 1  
## [2,] 2  
## [3,] 3  
## [4,] 4  
## [5,] 5  
## [6,] 6  
## [7,] 8  
## [8,] 9  
## [9,] 10  
## [10,] 11  
## [11,] 13  
## [12,] 14  
## [13,] 15  
## [14,] 16  
## [15,] 18  
## [16,] 19  
## [17,] 20  
## [18,] 21  
## [19,] 22  
## [20,] 23  
## [21,] 24  
## [22,] 25  
## [23,] 28  
## [24,] 29  
## [25,] 30  
## [26,] 32  
## [27,] 33  
## [28,] 36  
## [29,] 37  
## [30,] 38  
## [31,] 41  
## [32,] 42  
## [33,] 44  
## [34,] 46  
## [35,] 47  
## [36,] 50  
## [37,] 52

# Datos entrenamiento  
datos.entrenamiento <- datos[entrena, ] # [renglones, columna]  
kable(datos.entrenamiento, caption = "Datos de entrenamiento")

Datos de entrenamiento

|  | genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado | estado.01 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | MASCULINO | SOLTERO | 25 | 80 | 90 | 70 | 80.00 | BUENO | 90.0 | FELIZ | 1 |
| 2 | FEMENINO | CASADO | 35 | 50 | 80 | 60 | 70.00 | MALO | 30.0 | NO FELIZ | 0 |
| 3 | MASCULINO | DIVORCIADO | 45 | 70 | 78 | 80 | 40.00 | REGULAR | 70.0 | FELIZ | 1 |
| 4 | FEMENINO | VIUDO | 54 | 50 | 80 | 60 | 80.00 | BUENO | 20.0 | NO FELIZ | 0 |
| 5 | MASCULINO | CASADO | 52 | 40 | 50 | 60 | 70.00 | BUENO | 60.0 | NO FELIZ | 0 |
| 6 | FEMENINO | SOLTERO | 28 | 50 | 60 | 54 | 60.00 | MALO | 50.0 | NO FELIZ | 0 |
| 8 | FEMENINO | DIVORCIADO | 32 | 60 | 80 | 30 | 50.00 | REGULAR | 20.0 | NO FELIZ | 0 |
| 9 | MASCULINO | CASADO | 35 | 70 | 60 | 72 | 60.00 | BUENO | 70.0 | FELIZ | 1 |
| 10 | FEMENINO | SOLTERO | 29 | 80 | 80 | 90 | 60.00 | MALO | 80.0 | FELIZ | 1 |
| 11 | MASCULINO | DIVORCIADO | 45 | 60 | 60 | 70 | 50.00 | REGULAR | 90.0 | FELIZ | 1 |
| 13 | MASCULINO | CASADO | 26 | 50 | 45 | 80 | 60.00 | MALO | 20.0 | NO FELIZ | 0 |
| 14 | FEMENINO | SOLTERO | 34 | 60 | 40 | 50 | 80.00 | BUENO | 65.9 | FELIZ | 1 |
| 15 | MASCULINO | DIVORCIADO | 42 | 50 | 65 | 56 | 62.58 | REGULAR | 33.5 | NO FELIZ | 0 |
| 16 | FEMENINO | VIUDO | 35 | 80 | 70 | 20 | 20.00 | MALO | 20.5 | NO FELIZ | 0 |
| 18 | FEMENINO | SOLTERO | 34 | 54 | 80 | 56 | 60.00 | REGULAR | 55.0 | FELIZ | 1 |
| 19 | MASCULINO | DIVORCIADO | 34 | 60 | 70 | 80 | 50.00 | MALO | 100.0 | FELIZ | 1 |
| 20 | FEMENINO | SOLTERO | 32 | 40 | 50 | 80 | 90.00 | BUENO | 95.0 | FELIZ | 1 |
| 21 | MASCULINO | SOLTERO | 29 | 50 | 60 | 60 | 80.00 | MALO | 70.0 | FELIZ | 1 |
| 22 | FEMENINO | SOLTERO | 26 | 60 | 60 | 60 | 60.00 | BUENO | 60.0 | FELIZ | 1 |
| 23 | MASCULINO | CASADO | 45 | 60 | 60 | 60 | 60.00 | BUENO | 50.0 | FELIZ | 1 |
| 24 | FEMENINO | VIUDO | 45 | 50 | 60 | 60 | 30.00 | REGULAR | 35.0 | NO FELIZ | 0 |
| 25 | MASCULINO | CASADO | 28 | 50 | 40 | 80 | 30.00 | MALO | 30.0 | NO FELIZ | 0 |
| 28 | FEMENINO | CASADO | 40 | 45 | 50 | 40 | 90.00 | MALO | 60.0 | NO FELIZ | 0 |
| 29 | MASCULINO | SOLTERO | 41 | 60 | 60 | 60 | 60.00 | BUENO | 49.0 | FELIZ | 1 |
| 30 | FEMENINO | VIUDO | 38 | 80 | 70 | 30 | 40.00 | BUENO | 45.0 | NO FELIZ | 0 |
| 32 | FEMENINO | CASADO | 38 | 60 | 80 | 90 | 80.00 | BUENO | 60.0 | FELIZ | 1 |
| 33 | FEMENINO | CASADO | 37 | 80 | 60 | 70 | 50.00 | BUENO | 60.0 | FELIZ | 1 |
| 36 | FEMENINO | CASADO | 43 | 95 | 80 | 90 | 90.00 | BUENO | 80.0 | FELIZ | 1 |
| 37 | MASCULINO | CASADO | 55 | 70 | 70 | 65 | 89.00 | MALO | 75.0 | FELIZ | 1 |
| 38 | MASCULINO | SOLTERO | 45 | 65 | 70 | 45 | 65.00 | BUENO | 45.0 | FELIZ | 1 |
| 41 | MASCULINO | SOLTERO | 39 | 60 | 80 | 48 | 50.00 | BUENO | 60.0 | FELIZ | 1 |
| 42 | MASCULINO | SOLTERO | 42 | 90 | 80 | 50 | 100.00 | REGULAR | 80.0 | FELIZ | 1 |
| 44 | MASCULINO | VIUDO | 88 | 65 | 66 | 89 | 87.00 | REGULAR | 45.0 | NO FELIZ | 0 |
| 46 | MASCULINO | SOLTERO | 80 | 80 | 80 | 90 | 90.00 | BUENO | 90.0 | FELIZ | 1 |
| 47 | MASCULINO | SOLTERO | 39 | 50 | 80 | 90 | 80.00 | REGULAR | 90.0 | FELIZ | 1 |
| 50 | MASCULINO | SOLTERO | 39 | 60 | 80 | 48 | 50.00 | REGULAR | 50.0 | NO FELIZ | 0 |
| 52 | FEMENINO | SOLTERO | 32 | 70 | 80 | 80 | 80.00 | BUENO | 80.0 | FELIZ | 1 |

## Datos de validación

# Datos validación  
datos.validacion <- datos[-entrena, ] # Los aue no son de enrenamiento  
kable(datos.validacion, caption = "Datos de validación")

Datos de validación

|  | genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado | estado.01 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | MASCULINO | VIUDO | 56 | 71.5 | 60.8 | 86.9 | 70.6 | MALO | 70.0 | FELIZ | 1 |
| 12 | FEMENINO | VIUDO | 48 | 60.0 | 50.0 | 50.0 | 45.5 | MALO | 70.0 | NO FELIZ | 0 |
| 17 | MASCULINO | CASADO | 48 | 50.0 | 50.0 | 50.0 | 50.0 | BUENO | 49.5 | FELIZ | 1 |
| 26 | FEMENINO | DIVORCIADO | 32 | 80.0 | 70.0 | 40.0 | 40.0 | MALO | 40.0 | NO FELIZ | 0 |
| 27 | MASCULINO | SOLTERO | 36 | 65.0 | 60.0 | 62.0 | 87.0 | REGULAR | 56.5 | FELIZ | 1 |
| 31 | MASCULINO | CASADO | 36 | 85.0 | 80.0 | 90.0 | 50.0 | BUENO | 80.0 | FELIZ | 1 |
| 34 | FEMENINO | SOLTERO | 40 | 60.0 | 80.0 | 40.0 | 50.0 | REGULAR | 30.0 | NO FELIZ | 0 |
| 35 | MASCULINO | CASADO | 40 | 60.0 | 70.0 | 50.0 | 30.0 | BUENO | 50.0 | FELIZ | 1 |
| 39 | FEMENINO | CASADO | 41 | 24.0 | 57.0 | 33.0 | 71.0 | BUENO | 100.0 | NO FELIZ | 0 |
| 40 | FEMENINO | SOLTERO | 40 | 80.0 | 80.0 | 90.0 | 30.0 | REGULAR | 50.0 | FELIZ | 1 |
| 43 | FEMENINO | SOLTERO | 45 | 90.0 | 95.0 | 90.0 | 100.0 | BUENO | 90.0 | FELIZ | 1 |
| 45 | MASCULINO | CASADO | 54 | 90.0 | 90.0 | 90.0 | 100.0 | BUENO | 75.0 | FELIZ | 1 |
| 48 | FEMENINO | SOLTERO | 40 | 30.0 | 60.0 | 80.0 | 50.0 | REGULAR | 50.0 | NO FELIZ | 0 |
| 49 | FEMENINO | SOLTERO | 49 | 80.0 | 80.0 | 80.0 | 80.0 | BUENO | 70.0 | FELIZ | 1 |
| 51 | FEMENINO | SOLTERO | 30 | 100.0 | 100.0 | 100.0 | 100.0 | BUENO | 100.0 | NO FELIZ | 0 |

# Construir modelos

Con los datos de entrenamiento \* modelo1 = Regresión Logística \* modelo2 = Arbol de Regresión

## Modelos de Regresion Logística

modelo1 <- glm(data = datos.entrenamiento, formula = estado.01 ~ genero + esto.civil + edad + satisfaccion.laboral + satisfaccion.profesional + vida.familiar + vida.social + salud + dinero, family = "binomial") # el . significa todas las variables

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(modelo1)

##   
## Call:  
## glm(formula = estado.01 ~ genero + esto.civil + edad + satisfaccion.laboral +   
## satisfaccion.profesional + vida.familiar + vida.social +   
## salud + dinero, family = "binomial", data = datos.entrenamiento)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.825e-05 -2.110e-08 2.110e-08 2.110e-08 1.842e-05   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.326e+02 6.515e+05 0.000 1.000  
## generoMASCULINO -1.077e+01 7.278e+04 0.000 1.000  
## esto.civilDIVORCIADO 1.385e+01 2.019e+05 0.000 1.000  
## esto.civilSOLTERO 9.768e-02 1.614e+05 0.000 1.000  
## esto.civilVIUDO -4.821e+01 3.101e+05 0.000 1.000  
## edad -2.973e+00 8.091e+03 0.000 1.000  
## satisfaccion.laboral 2.303e+00 4.192e+03 0.001 1.000  
## satisfaccion.profesional 5.020e-01 4.300e+03 0.000 1.000  
## vida.familiar 1.342e+00 6.906e+03 0.000 1.000  
## vida.social 1.852e+00 4.130e+03 0.000 1.000  
## saludMALO -7.657e+01 1.055e+05 -0.001 0.999  
## saludREGULAR -3.837e+01 9.452e+04 0.000 1.000  
## dinero 7.932e-01 3.808e+03 0.000 1.000  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 4.9082e+01 on 36 degrees of freedom  
## Residual deviance: 1.8725e-09 on 24 degrees of freedom  
## AIC: 26  
##   
## Number of Fisher Scoring iterations: 25

## Predicciones

Con los datos de validación

### Predicciones con el modelo logístico

predicciones1 <- predict(object = modelo1,newdata = datos.validacion, se.fit = TRUE)  
predicciones1

## $fit  
## 7 12 17 26 27 31 34   
## -36.52048 -129.86755 -46.81210 -11.52163 80.35405 162.39766 -41.30946   
## 35 39 40 43 45 48 49   
## -26.60262 -15.45162 50.65058 266.10773 214.06945 -50.92787 157.32540   
## 51   
## 357.59619   
##   
## $se.fit  
## 7 12 17 26 27 31 34 35   
## 439303.50 221628.14 102522.87 188166.78 172296.64 205542.27 95795.94 162537.98   
## 39 40 43 45 48 49 51   
## 378048.09 302608.49 332563.26 285269.17 232332.36 279954.58 365652.36   
##   
## $residual.scale  
## [1] 1

### Las probabilidades

# Mediante la función logit se transforman los a probabilidades.  
predicciones1\_prob <- exp(predicciones1$fit) / (1 + exp(predicciones1$fit))

### Tabla a comprar 1

Construir un data.frame de los datos de validación con las probabilidaes y el valor resultante 0 o 1 dependiendo de si es por encima del 50% la probabilidad

tabla.comparar1 <- data.frame(datos.validacion, predicciones1\_prob)  
  
tabla.comparar1 <- tabla.comparar1 %>%  
 mutate(estado.pred = if\_else(predicciones1\_prob < 0.50, 0, 1))  
kable(tabla.comparar1)

|  | genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado | estado.01 | predicciones1\_prob | estado.pred |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | MASCULINO | VIUDO | 56 | 71.5 | 60.8 | 86.9 | 70.6 | MALO | 70.0 | FELIZ | 1 | 0.0e+00 | 0 |
| 12 | FEMENINO | VIUDO | 48 | 60.0 | 50.0 | 50.0 | 45.5 | MALO | 70.0 | NO FELIZ | 0 | 0.0e+00 | 0 |
| 17 | MASCULINO | CASADO | 48 | 50.0 | 50.0 | 50.0 | 50.0 | BUENO | 49.5 | FELIZ | 1 | 0.0e+00 | 0 |
| 26 | FEMENINO | DIVORCIADO | 32 | 80.0 | 70.0 | 40.0 | 40.0 | MALO | 40.0 | NO FELIZ | 0 | 9.9e-06 | 0 |
| 27 | MASCULINO | SOLTERO | 36 | 65.0 | 60.0 | 62.0 | 87.0 | REGULAR | 56.5 | FELIZ | 1 | 1.0e+00 | 1 |
| 31 | MASCULINO | CASADO | 36 | 85.0 | 80.0 | 90.0 | 50.0 | BUENO | 80.0 | FELIZ | 1 | 1.0e+00 | 1 |
| 34 | FEMENINO | SOLTERO | 40 | 60.0 | 80.0 | 40.0 | 50.0 | REGULAR | 30.0 | NO FELIZ | 0 | 0.0e+00 | 0 |
| 35 | MASCULINO | CASADO | 40 | 60.0 | 70.0 | 50.0 | 30.0 | BUENO | 50.0 | FELIZ | 1 | 0.0e+00 | 0 |
| 39 | FEMENINO | CASADO | 41 | 24.0 | 57.0 | 33.0 | 71.0 | BUENO | 100.0 | NO FELIZ | 0 | 2.0e-07 | 0 |
| 40 | FEMENINO | SOLTERO | 40 | 80.0 | 80.0 | 90.0 | 30.0 | REGULAR | 50.0 | FELIZ | 1 | 1.0e+00 | 1 |
| 43 | FEMENINO | SOLTERO | 45 | 90.0 | 95.0 | 90.0 | 100.0 | BUENO | 90.0 | FELIZ | 1 | 1.0e+00 | 1 |
| 45 | MASCULINO | CASADO | 54 | 90.0 | 90.0 | 90.0 | 100.0 | BUENO | 75.0 | FELIZ | 1 | 1.0e+00 | 1 |
| 48 | FEMENINO | SOLTERO | 40 | 30.0 | 60.0 | 80.0 | 50.0 | REGULAR | 50.0 | NO FELIZ | 0 | 0.0e+00 | 0 |
| 49 | FEMENINO | SOLTERO | 49 | 80.0 | 80.0 | 80.0 | 80.0 | BUENO | 70.0 | FELIZ | 1 | 1.0e+00 | 1 |
| 51 | FEMENINO | SOLTERO | 30 | 100.0 | 100.0 | 100.0 | 100.0 | BUENO | 100.0 | NO FELIZ | 0 | 1.0e+00 | 1 |

### Evaluar con una matriz de confusión

Evaluar las métricas de:

* Eficiencia Exactitud: Accuracy.
* Precisión : Precision.

### Generar la matriz de confusión

Antes se factorizan las variables estado.01 y estado.pred

tabla.comparar1$estado.01 = as.factor(tabla.comparar1$estado.01)  
tabla.comparar1$estado.pred = as.factor(tabla.comparar1$estado.pred)

Se utiliza confusionMatrix de la librería caret

matriz1 <- confusionMatrix(tabla.comparar1$estado.01, tabla.comparar1$estado.pred)  
matriz1

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 5 1  
## 1 3 6  
##   
## Accuracy : 0.7333   
## 95% CI : (0.449, 0.9221)  
## No Information Rate : 0.5333   
## P-Value [Acc > NIR] : 0.09638   
##   
## Kappa : 0.4737   
##   
## Mcnemar's Test P-Value : 0.61708   
##   
## Sensitivity : 0.6250   
## Specificity : 0.8571   
## Pos Pred Value : 0.8333   
## Neg Pred Value : 0.6667   
## Prevalence : 0.5333   
## Detection Rate : 0.3333   
## Detection Prevalence : 0.4000   
## Balanced Accuracy : 0.7411   
##   
## 'Positive' Class : 0   
##

## Arbol de clasificación

### Ajustar prámatros para el árbol

Ajustes para ramificaciones y tope del árbol

ajustes <- rpart.control(minsplit = 2,  
 minbucket = round(3 / 2),  
 maxdepth = 4,  
 cp = 0)

### Crear el modelo de árbol de clasificación

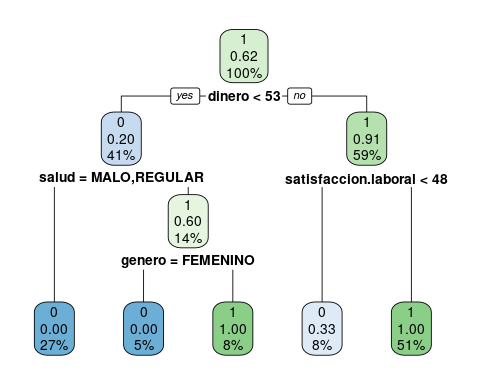
Crear el modelo de árbol de clasificación con la función rpart de la librería part.

modelo2 <- rpart(data = datos.entrenamiento, formula = estado.01 ~ genero + esto.civil + edad + satisfaccion.laboral + satisfaccion.profesional + vida.familiar + vida.social + salud + dinero, method = "class", control = ajustes)

## Visualizando el árbol

Se visualiza el árbol con el modelo2 de árbol de clasificación.

rpart.plot(modelo2)



## Predicciones con el árbol de clasificación

predicciones2 <- predict(object = modelo2, newdata = datos.validacion)

### Tabla a comprar 2

Construir un data.frame de los datos de validación con las predicciones

tabla.comparar2 <- data.frame(datos.validacion, predicciones2)  
  
tabla.comparar2 <- tabla.comparar2 %>%  
 mutate(estado.pred = if\_else( X1 < 0.50, 0, 1))  
kable(tabla.comparar2)

|  | genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado | estado.01 | X0 | X1 | estado.pred |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | MASCULINO | VIUDO | 56 | 71.5 | 60.8 | 86.9 | 70.6 | MALO | 70.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 12 | FEMENINO | VIUDO | 48 | 60.0 | 50.0 | 50.0 | 45.5 | MALO | 70.0 | NO FELIZ | 0 | 0.0000000 | 1.0000000 | 1 |
| 17 | MASCULINO | CASADO | 48 | 50.0 | 50.0 | 50.0 | 50.0 | BUENO | 49.5 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 26 | FEMENINO | DIVORCIADO | 32 | 80.0 | 70.0 | 40.0 | 40.0 | MALO | 40.0 | NO FELIZ | 0 | 1.0000000 | 0.0000000 | 0 |
| 27 | MASCULINO | SOLTERO | 36 | 65.0 | 60.0 | 62.0 | 87.0 | REGULAR | 56.5 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 31 | MASCULINO | CASADO | 36 | 85.0 | 80.0 | 90.0 | 50.0 | BUENO | 80.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 34 | FEMENINO | SOLTERO | 40 | 60.0 | 80.0 | 40.0 | 50.0 | REGULAR | 30.0 | NO FELIZ | 0 | 1.0000000 | 0.0000000 | 0 |
| 35 | MASCULINO | CASADO | 40 | 60.0 | 70.0 | 50.0 | 30.0 | BUENO | 50.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 39 | FEMENINO | CASADO | 41 | 24.0 | 57.0 | 33.0 | 71.0 | BUENO | 100.0 | NO FELIZ | 0 | 0.6666667 | 0.3333333 | 0 |
| 40 | FEMENINO | SOLTERO | 40 | 80.0 | 80.0 | 90.0 | 30.0 | REGULAR | 50.0 | FELIZ | 1 | 1.0000000 | 0.0000000 | 0 |
| 43 | FEMENINO | SOLTERO | 45 | 90.0 | 95.0 | 90.0 | 100.0 | BUENO | 90.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 45 | MASCULINO | CASADO | 54 | 90.0 | 90.0 | 90.0 | 100.0 | BUENO | 75.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 48 | FEMENINO | SOLTERO | 40 | 30.0 | 60.0 | 80.0 | 50.0 | REGULAR | 50.0 | NO FELIZ | 0 | 1.0000000 | 0.0000000 | 0 |
| 49 | FEMENINO | SOLTERO | 49 | 80.0 | 80.0 | 80.0 | 80.0 | BUENO | 70.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 51 | FEMENINO | SOLTERO | 30 | 100.0 | 100.0 | 100.0 | 100.0 | BUENO | 100.0 | NO FELIZ | 0 | 0.0000000 | 1.0000000 | 1 |

kable(tabla.comparar2)

|  | genero | esto.civil | edad | satisfaccion.laboral | satisfaccion.profesional | vida.familiar | vida.social | salud | dinero | estado | estado.01 | X0 | X1 | estado.pred |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | MASCULINO | VIUDO | 56 | 71.5 | 60.8 | 86.9 | 70.6 | MALO | 70.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 12 | FEMENINO | VIUDO | 48 | 60.0 | 50.0 | 50.0 | 45.5 | MALO | 70.0 | NO FELIZ | 0 | 0.0000000 | 1.0000000 | 1 |
| 17 | MASCULINO | CASADO | 48 | 50.0 | 50.0 | 50.0 | 50.0 | BUENO | 49.5 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 26 | FEMENINO | DIVORCIADO | 32 | 80.0 | 70.0 | 40.0 | 40.0 | MALO | 40.0 | NO FELIZ | 0 | 1.0000000 | 0.0000000 | 0 |
| 27 | MASCULINO | SOLTERO | 36 | 65.0 | 60.0 | 62.0 | 87.0 | REGULAR | 56.5 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 31 | MASCULINO | CASADO | 36 | 85.0 | 80.0 | 90.0 | 50.0 | BUENO | 80.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 34 | FEMENINO | SOLTERO | 40 | 60.0 | 80.0 | 40.0 | 50.0 | REGULAR | 30.0 | NO FELIZ | 0 | 1.0000000 | 0.0000000 | 0 |
| 35 | MASCULINO | CASADO | 40 | 60.0 | 70.0 | 50.0 | 30.0 | BUENO | 50.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 39 | FEMENINO | CASADO | 41 | 24.0 | 57.0 | 33.0 | 71.0 | BUENO | 100.0 | NO FELIZ | 0 | 0.6666667 | 0.3333333 | 0 |
| 40 | FEMENINO | SOLTERO | 40 | 80.0 | 80.0 | 90.0 | 30.0 | REGULAR | 50.0 | FELIZ | 1 | 1.0000000 | 0.0000000 | 0 |
| 43 | FEMENINO | SOLTERO | 45 | 90.0 | 95.0 | 90.0 | 100.0 | BUENO | 90.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 45 | MASCULINO | CASADO | 54 | 90.0 | 90.0 | 90.0 | 100.0 | BUENO | 75.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 48 | FEMENINO | SOLTERO | 40 | 30.0 | 60.0 | 80.0 | 50.0 | REGULAR | 50.0 | NO FELIZ | 0 | 1.0000000 | 0.0000000 | 0 |
| 49 | FEMENINO | SOLTERO | 49 | 80.0 | 80.0 | 80.0 | 80.0 | BUENO | 70.0 | FELIZ | 1 | 0.0000000 | 1.0000000 | 1 |
| 51 | FEMENINO | SOLTERO | 30 | 100.0 | 100.0 | 100.0 | 100.0 | BUENO | 100.0 | NO FELIZ | 0 | 0.0000000 | 1.0000000 | 1 |

### Evaluar con una matriz de confusión

Evaluar las métricas de:

* Eficiencia Exactitud: Accuracy.
* Precisión : Precision.

### Generar la matriz de confusión

Antes se factorizan las variables estado.01 y estado.pred

tabla.comparar2$estado.01 = as.factor(tabla.comparar1$estado.01)  
tabla.comparar2$estado.pred = as.factor(tabla.comparar2$estado.pred)

Se utiliza *confusionMatrix* de la librería caret

matriz2 <- confusionMatrix(tabla.comparar2$estado.01, tabla.comparar2$estado.pred)  
matriz2

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 4 2  
## 1 1 8  
##   
## Accuracy : 0.8   
## 95% CI : (0.5191, 0.9567)  
## No Information Rate : 0.6667   
## P-Value [Acc > NIR] : 0.2092   
##   
## Kappa : 0.5714   
##   
## Mcnemar's Test P-Value : 1.0000   
##   
## Sensitivity : 0.8000   
## Specificity : 0.8000   
## Pos Pred Value : 0.6667   
## Neg Pred Value : 0.8889   
## Prevalence : 0.3333   
## Detection Rate : 0.2667   
## Detection Prevalence : 0.4000   
## Balanced Accuracy : 0.8000   
##   
## 'Positive' Class : 0   
##

## Evaluar las métricas

kable(matriz1$table, caption = "Matriz de confusión del modelo de regresión logística")

Matriz de confusión del modelo de regresión logística

|  | 0 | 1 |
| --- | --- | --- |
| 0 | 5 | 1 |
| 1 | 3 | 6 |

kable(matriz2$table, caption = "Matriz de confusión del modelo de árbol de clasificación")

Matriz de confusión del modelo de árbol de clasificación

|  | 0 | 1 |
| --- | --- | --- |
| 0 | 4 | 2 |
| 1 | 1 | 8 |

kable(matriz1$overall, caption = "Métricas del del modelo de regresión logística")

Métricas del del modelo de regresión logística

|  | x |
| --- | --- |
| Accuracy | 0.7333333 |
| Kappa | 0.4736842 |
| AccuracyLower | 0.4489968 |
| AccuracyUpper | 0.9221285 |
| AccuracyNull | 0.5333333 |
| AccuracyPValue | 0.0963762 |
| McnemarPValue | 0.6170751 |

kable(matriz2$overall, caption = "Métricas del modelo de árbol de clasificación")

Métricas del modelo de árbol de clasificación

|  | x |
| --- | --- |
| Accuracy | 0.8000000 |
| Kappa | 0.5714286 |
| AccuracyLower | 0.5191089 |
| AccuracyUpper | 0.9566880 |
| AccuracyNull | 0.6666667 |
| AccuracyPValue | 0.2092402 |
| McnemarPValue | 1.0000000 |

Se puede observar que el modelo de árbol de clasificación le falló a 3 observaciones mientras que el modelo de regresión logística le falla en 4 ocasiones con respecto a los datos de validación.

Se puede interpretar que en términos de accuracy (eficiencia o exactitud) el modelo de árbol de clasificación es más eficiente que el de regresión logística con estos datos.

# Predicciones con datos nuevos

¿Cual sería un dato nuevo?

genero = 'FEMENINO'  
esto.civil = 'SOLTERO'  
edad = 52  
satisfaccion.laboral <- 40  
satisfaccion.profesional <- 60  
vida.familiar <- 40  
vida.social <- 60  
salud <- 'REGULAR'  
dinero <- 40  
  
datos.nuevos <- data.frame(genero, esto.civil, edad, satisfaccion.laboral, satisfaccion.profesional, vida.familiar, vida.social, salud, dinero)  
datos.nuevos

## genero esto.civil edad satisfaccion.laboral satisfaccion.profesional  
## 1 FEMENINO SOLTERO 52 40 60  
## vida.familiar vida.social salud dinero  
## 1 40 60 REGULAR 40

predicciones.nuevas <- predict(object = modelo2, newdata = datos.nuevos)  
predicciones.nuevas

## 0 1  
## 1 1 0

paste("La PREDICCIÓN O CLASIFICACIÓN de esta persona con estas características es: ", ifelse(predicciones.nuevas[,2] == 0, "NO FELIZ", "FELIZ") )

## [1] "La PREDICCIÓN O CLASIFICACIÓN de esta persona con estas características es: NO FELIZ"

# Interpretar el caso

# Bibliografía