Actividad Integradora 2

Erika Martínez Meneses

2024-11-19

Utiliza los archivos del Titanic para detectar cuáles fueron las principales características que de las personas que sobrevivieron y elabora en modelo de predicción de sobrevivencia o no en el Titanic. Utiliza en las siguientes bases de datos:

- Base de datos del Titanic: Titanic
- Base de datos de prueba: Titanic_test

Las variables para la base de datos son las siguientes (excluye aquellas que no sean de interés para el análisis):

- *Name:* Nombre del pasajero
- *PassengerId:* Ids del pasajero
- *Survived:* Si sobrevivió o no (No = 0, Sí = 1)
- *Ticket:* Número de ticket
- *Cabin:* Cabina en la que viajó
- *Pclass:* Clase en la que viajó (1 = 1era, 2 = 2da, 3 = 3ra)
- *Sex:* Masculino o Femenino (male/female)
- *Age:* Edad
- *SibSp:* Número de hermanos/conyuge a bordo
- *Parch:* Número de padres/hijos a bordo
- *Fare:* Tarifa que pagó
- *Embarked:* Puerto de embarcación (C = Cherbourg, Q = Queenstown, S = Southampton)

Sigue los siguientes pasos:

1. Prepara la base de datos Titanic:

Librerías

```
# Cargamos todas las librería en la lista "librerias"
librerias = c('tidyverse','broom','ISLR','GGally','modelr','cowplot','rla
ng','modelr','tibble','Metrics','mice','visdat',"caret", "pROC")

for (lib in librerias){
  library(lib,character.only=TRUE)}
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyvers
e 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.1     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conf
licts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force
all conflicts to become errors
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
##
##
## Attaching package: 'modelr'
##
##
## The following object is masked from 'package:broom':
##
##     bootstrap
##
##
##
## Attaching package: 'cowplot'
##
##
## The following object is masked from 'package:lubridate':
##
##     stamp
##
##
##
## Attaching package: 'rlang'
##
```

```
##
## The following objects are masked from 'package:purrr':
##
##     %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
##     flatten_raw, invoke, splice

## Warning: package 'Metrics' was built under R version 4.4.2

##
## Attaching package: 'Metrics'
##
## The following object is masked from 'package:rlang':
##
##     ll
##
## The following objects are masked from 'package:modelr':
##
##     mae, mape, mse, rmse

## Warning: package 'mice' was built under R version 4.4.2

##
## Attaching package: 'mice'
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following objects are masked from 'package:base':
##
##     cbind, rbind

## Warning: package 'visdat' was built under R version 4.4.2

## Warning: package 'caret' was built under R version 4.4.2

## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following objects are masked from 'package:Metrics':
##
##     precision, recall
##
## The following object is masked from 'package:purrr':
##
##     lift

## Warning: package 'pROC' was built under R version 4.4.2

## Type 'citation("pROC")' for a citation.
##
```

```
## Attaching package: 'pROC'
##
## The following object is masked from 'package:Metrics':
##
##     auc
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
file.choose()
```

```
## [1] "C:\\Users\\erika\\Documents\\Agos-Dic2024\\Estadística\\Titanic_test.csv"
```

```
file.choose()
```

```
## [1] "C:\\Users\\erika\\Documents\\Agos-Dic2024\\Estadística\\Titanic.csv"
```

```
titanic_original <- read.csv("C:\\Users\\erika\\Documents\\Agos-Dic2024\\Estadística\\Titanic.csv")
titanic_test <- read.csv("C:\\Users\\erika\\Documents\\Agos-Dic2024\\Estadística\\Titanic_test.csv")
```

Para la selección de columnas eliminamos aquellas que no consideramos relevantes y convertimos las variables categóricas a factores, las varaibles que convertimos a favores

```
# Eliminar variables:
titanic_original <- titanic_original[,c(-4,-9,-11)]

#Transformar a factores:
for(var in c('Survived','Pclass','Embarked','Sex'))
  titanic_original[,var] <-as.factor(titanic_original[,var])
```

Analiza los datos faltantes

Detectar si hay espacios vacíos en lugar de datos:

```
V = matrix(NA,ncol=1,nrow=9)
for(i in c(1:9)){
  V[i,] <- sum(with(titanic_original,titanic_original[,i])=="")}
V
```

```
##       [,1]
##  [1,]    0
##  [2,]    0
##  [3,]    0
##  [4,]    0
##  [5,]   NA
##  [6,]    0
```

```
##  [7,]    0
##  [8,]   NA
##  [9,]   NA
```

Ninguna variable contiene espacios vacíos, pero las variables 5 (Age), 8 (Fare) y 9 (Embarked) tienen datos faltantes. Contamis cuántos datos faltantes tienen las columnas.

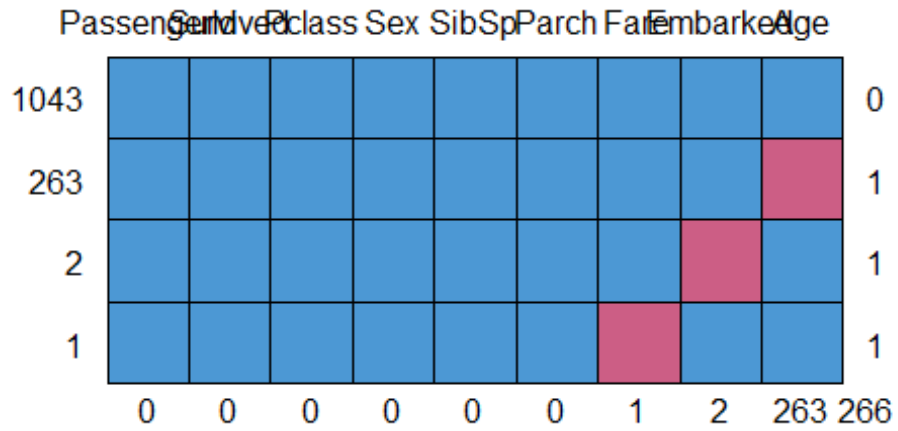
```
missing_summary <- colSums(is.na(titanic_original))
missing_summary
```

```
## PassengerId    Survived      Pclass         Sex         Age       SibS
p
##           0           0           0           0         263
0
##       Parch        Fare    Embarked
##           0           1           2
```

Podemos observar que Fare tiene 1 dato faltante, Embarked 2 y en Edad hay múltiples datos faltantes (263) que representan el 20% de los datos.

Patrón de los datos faltantes

```
md.pattern(titanic_original)
```

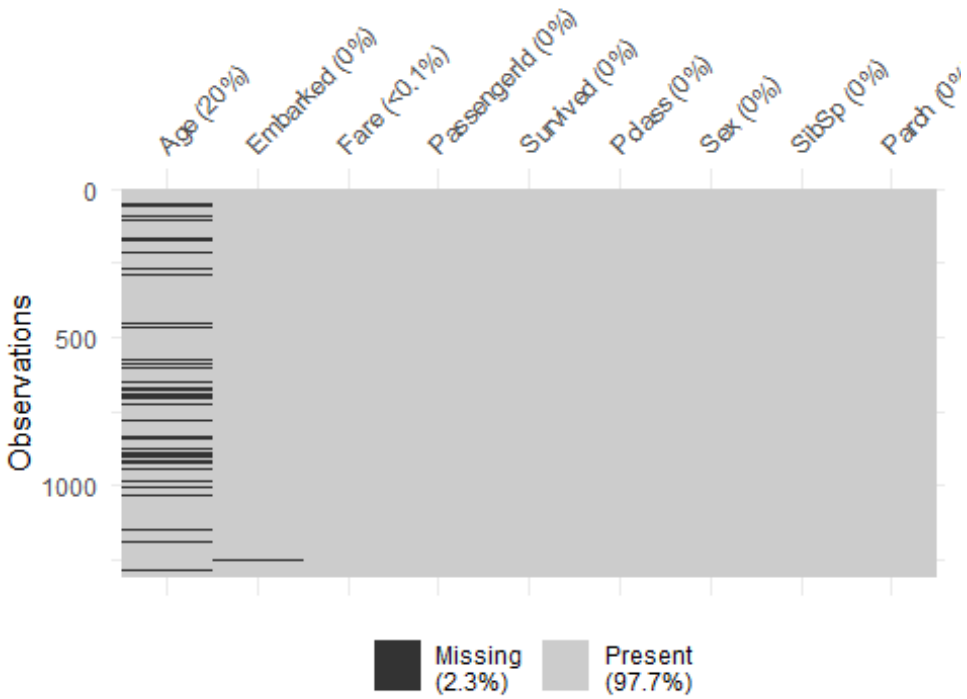



```
##      PassengerId Survived Pclass Sex SibSp Parch Fare Embarked Age
## 1043           1        1      1   1     1     1    1        1   1   0
## 263            1        1      1   1     1     1    1        1   0   1
```

```
## 2              1        1      1   1     1     1    1        0   1   1
## 1              1        1      1   1     1     1    0        1   1   1
##                0        0      0   0     0     0    1        2 263 266
```

Gracias a pattern podemos observar que todos los datos faltantes son de distintos pasajeros (observaciones), por lo tanto, si se eliminan los NA, se eliminarían 266 observaciones y nos quedaríamos con 1043 observaciones.

```
vis_miss(titanic_original,sort_miss = TRUE)
```



Con esta gráfica podemos observar el porcentaje de datos faltantes por cada columna. Identificamos que Edad tiene el 20% de datos faltantes.

Medidas con datos faltantes

```
summary(titanic_original[,-1])
```

```
##  Survived Pclass      Sex           Age            SibSp           Pa
rch
##  0:815    1:323   female:466   Min.   : 0.17   Min.   :0.0000   Min.
:0.000
##  1:494    2:277   male  :843   1st Qu.:21.00   1st Qu.:0.0000   1st Qu
.:0.000
##           3:709                Median :28.00   Median :0.0000   Median
:0.000
##                                Mean   :29.88   Mean   :0.4989   Mean
:0.385
##                                3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu
```

```
.:0.000
##                                 Max.   :80.00   Max.   :8.0000   Max.
:9.000
##                                 NA's   :263
##       Fare          Embarked
##  Min.   :  0.000   C   :270
##  1st Qu.:  7.896   Q   :123
##  Median : 14.454   S   :914
##  Mean   : 33.295   NA's:  2
##  3rd Qu.: 31.275
##  Max.   :512.329
##  NA's   :1
```

Medidas sin datos faltantes

```
titanic = na.omit(titanic_original)
summary(titanic[,-1])
```

```
##  Survived Pclass      Sex           Age            SibSp
##  0:628    1:282   female:386   Min.   : 0.17   Min.   :0.0000
##  1:415    2:261   male  :657   1st Qu.:21.00   1st Qu.:0.0000
##           3:500                Median :28.00   Median :0.0000
##                                Mean   :29.81   Mean   :0.5043
##                                3rd Qu.:39.00   3rd Qu.:1.0000
##                                Max.   :80.00   Max.   :8.0000
##      Parch             Fare        Embarked
##  Min.   :0.0000   Min.   :  0.00   C:212
##  1st Qu.:0.0000   1st Qu.:  8.05   Q: 50
##  Median :0.0000   Median : 15.75   S:781
##  Mean   :0.4219   Mean   : 36.60
##  3rd Qu.:1.0000   3rd Qu.: 35.08
##  Max.   :6.0000   Max.   :512.33
```

Sobrevivientes

```
t2c = 100*prop.table(table(titanic_original[,2]))
t2s = 100*prop.table(table(titanic[,2]))
t2p = c(t2s[1]/t2c[1],t2s[2]/t2c[2])
t2 = data.frame(as.numeric(t2c),as.numeric(t2s),as.numeric(t2p))
row.names(t2) = c("Murió","Sobrevivió")
names(t2) = c("Con NA (%)","Sin NA (%)","Pérdida (prop)")
round(t2,2)
```

```
##            Con NA (%) Sin NA (%) Pérdida (prop)
## Murió           62.26      60.21           0.97
## Sobrevivió      37.74      39.79           1.05
```

Clase en que viajó

```
t3c = 100*prop.table(table(titanic_original[,3]))
t3s = 100*prop.table(table(titanic[,3]))
t3p = c(t3s[1]/t3c[1],t3s[2]/t3c[2],t3s[3]/t3c[3])
```

```
t3 = data.frame(as.numeric(t3c),as.numeric(t3s),as.numeric(t3p))
row.names(t3) = c("Primera","Segunda","Tercera")
names(t3) = c("Con NA (%)","Sin NA (%)","Pérdida (prop)")
round(t3,2)
```

```
##         Con NA (%) Sin NA (%) Pérdida (prop)
## Primera      24.68      27.04           1.10
## Segunda      21.16      25.02           1.18
## Tercera      54.16      47.94           0.89
```

Sexo

```
t4c = 100*prop.table(table(titanic_original[,4]))
t4s = 100*prop.table(table(titanic[,4]))
t4p = c(t4s[1]/t4c[1],t4s[2]/t4c[2])
t4 = data.frame(as.numeric(t4c),as.numeric(t4s),as.numeric(t4p))
row.names(t4) = c("Mujer","Hombre")
names(t4) = c("Con NA (%)","Sin NA (%)","Pérdida (prop)")
round(t4,2)
```

```
##        Con NA (%) Sin NA (%) Pérdida (prop)
## Mujer        35.6      37.01           1.04
## Hombre       64.4      62.99           0.98
```

Puerto de embarcación

```
t9c = 100*prop.table(table(titanic_original[,9]))
t9s = 100*prop.table(table(titanic[,9]))
t9p = c(t9s[1]/t9c[1],t9s[2]/t9c[2],t9s[3]/t9c[3])
t9 = data.frame(as.numeric(t9c),as.numeric(t9s),as.numeric(t9p))
row.names(t9) = c("Cherbourg","Queenstown","Southampton")
names(t9) = c("Con NA (%)","Sin NA (%)","Pérdida (prop)")
round(t9,2)
```

```
##             Con NA (%) Sin NA (%) Pérdida (prop)
## Cherbourg        20.66      20.33           0.98
## Queenstown        9.41       4.79           0.51
## Southampton      69.93      74.88           1.07
```

Se eliminaron los datos faltantes

Realiza un análisis descriptivo

```
summary(titanic)
```

```
##   PassengerId     Survived Pclass      Sex           Age       
##  Min.   :   1.0   0:628    1:282   female:386   Min.   : 0.17  
##  1st Qu.: 326.5   1:415    2:261   male  :657   1st Qu.:21.00  
##  Median : 662.0            3:500                 Median :28.00  
##  Mean   : 655.4                                  Mean   :29.81  
##  3rd Qu.: 973.5                                  3rd Qu.:39.00  
##  Max.   :1307.0                                  Max.   :80.00  
##      SibSp            Parch             Fare        Embarked  
```

```
##  Min.   :0.0000   Min.   :0.0000   Min.   :  0.00   C:212
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:  8.05   Q: 50
##  Median :0.0000   Median :0.0000   Median : 15.75   S:781
##  Mean   :0.5043   Mean   :0.4219   Mean   : 36.60
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.: 35.08
##  Max.   :8.0000   Max.   :6.0000   Max.   :512.33
```

¿Difieren las medidas con o sin datos faltantes? ¿cuáles son las variables que más se ven afectadas?

Sí, las medidas difieren ligeramente con y sin datos faltantes. Las variables más afectadas son Age, con 263 valores faltantes que reducen el tamaño de la muestra, aunque las medias y medianas cambian mínimamente, y Fare, que muestra un cambio más notable en la media (de 33.30 a 36.60) y la mediana (de 14.45 a 15.75). Las variables categóricas, como Survived y Embarked, mantienen proporciones similares, indicando un impacto menor en su distribución.

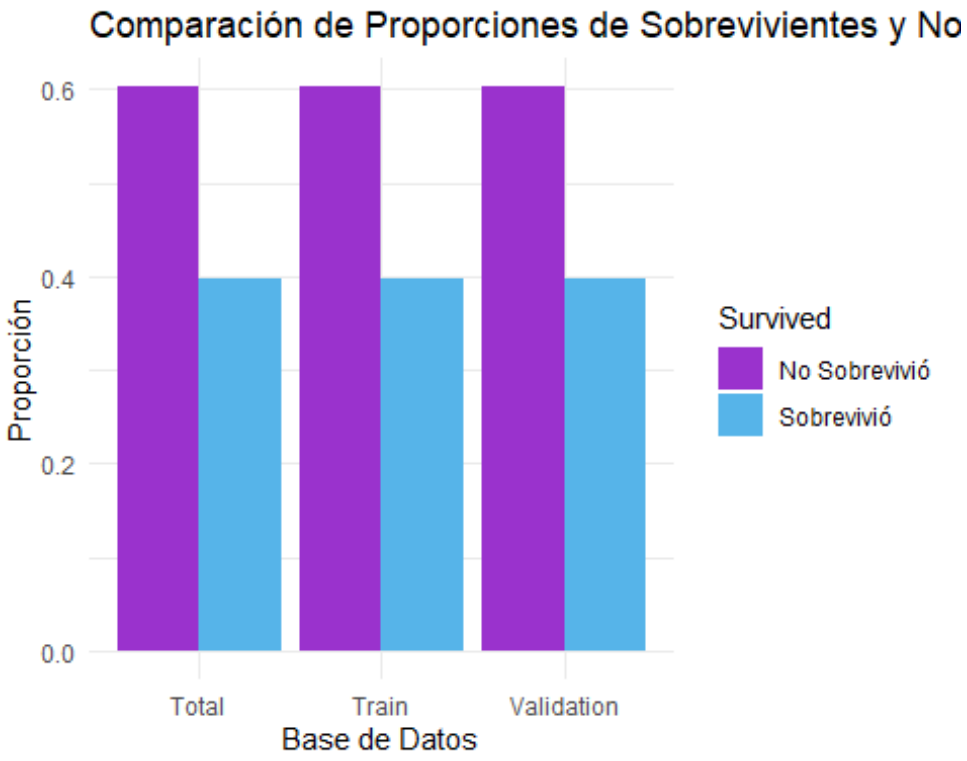
Haz una partición de los datos (70-30) para el entrenamiento y la validación. Revisa la proporción de sobrevivientes para la partición y la base original.

```
# Partición de los datos
set.seed(123)
train_index <- createDataPartition(titanic$Survived, p = 0.7, list = FALS
E)
titanic_train <- titanic[train_index, ]
titanic_valid <- titanic[-train_index, ]

total_prop <- prop.table(table(titanic$Survived))
train_prop <- prop.table(table(titanic_train$Survived))
valid_prop <- prop.table(table(titanic_valid$Survived))

prop_df <- data.frame(
  BaseDatos = rep(c("Total", "Train", "Validation"), each = 2),
  Survived = rep(c(0, 1), times = 3),
  Proportion = c(total_prop, train_prop, valid_prop)
)
prop_df
```

```
##    BaseDatos Survived Proportion
## 1      Total        0  0.6021093
## 2      Total        1  0.3978907
## 3      Train        0  0.6019152
## 4      Train        1  0.3980848
## 5 Validation        0  0.6025641
## 6 Validation        1  0.3974359
```

```
ggplot(prop_df, aes(x = BaseDatos, y = Proportion, fill = as.factor(Survi
ved))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Comparación de Proporciones de Sobrevivientes y No Sobrev
```

```
ivientes",
       x = "Base de Datos", y = "Proporción", fill = "Survived") +
  scale_fill_manual(values = c("#9A32CD", "#56B4E9"), labels = c("No Sobr
evivió", "Sobrevivió")) +
  theme_minimal()
```



Gracias a la tabla y a las gráficas podemos observar que la proporción de no sobrevientes se mantiene en las tres bases de datos.

2.Modelación

Con la base de datos de entrenamiento, se busca un modelo logístico para encontrar el mejor conjunto de predictores que auxilien a clasificar la dirección de cada observación. Se comienza con el modelo completo, incluyendo las variables categóricas (factores) y posteriormente se aplica el comando *step* para poder encontrar el mejor modelo.

Auxiliate del criterio de AIC para determinar cuál es el mejor modelo.

Generamos el modelo incial con todas las variables

```
A <- glm(Survived ~ Survived + Pclass + Sex +Age + SibSp + Parch + Fare +
Embarked, data = titanic_train, family = "binomial")
```

```
## Warning in model.matrix.default(mt, mf, contrasts): the response appea
red on
## the right-hand side and was dropped

## Warning in model.matrix.default(mt, mf, contrasts): problem with term
1 in
## model.matrix: no columns are assigned
```

Aplicamos una selección basada en AIC

step utiliza el criterio de Aikaike (AIC) para definir el mejor modelo, sin embargo también proporciona la desviación residual del modelo completo. Un menor AIC y una menor *Deviance* indicarán un mejor modelo.

```
model_step <- step(A, direction = "both", trace=1)

## Start:  AIC=564.23
## Survived ~ Survived + Pclass + Sex + Age + SibSp + Parch + Fare +
##     Embarked

## Warning in model.matrix.default(object, data = structure(list(Survived
=
## structure(c(2L, : the response appeared on the right-hand side and was
dropped

## Warning in model.matrix.default(object, data = structure(list(Survived
=
## structure(c(2L, : problem with term 1 in model.matrix: no columns are
assigned

##
## Step:  AIC=564.23
## Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## the response appeared on the right-hand side and was dropped

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## problem with term 8 in model.matrix: no columns are assigned

##             Df Deviance    AIC
## - Embarked  2   544.33 560.33
## - Parch     1   544.43 562.43
## - Fare      1   545.22 563.22
## <none>          544.23 564.23
## - SibSp     1   549.17 567.17
## - Age       1   555.12 573.12
## - Pclass    2   570.87 586.87
## - Sex       1   878.22 896.22
##
```

```
## Step:  AIC=560.33
## Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## the response appeared on the right-hand side and was dropped

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## problem with term 7 in model.matrix: no columns are assigned

##            Df Deviance    AIC
## - Parch     1   544.55 558.55
## - Fare      1   545.47 559.47
## <none>          544.33 560.33
## - SibSp     1   549.36 563.36
## + Embarked  2   544.23 564.23
## - Age       1   555.57 569.57
## - Pclass    2   572.79 584.79
## - Sex       1   884.39 898.39
##
## Step:  AIC=558.55
## Survived ~ Pclass + Sex + Age + SibSp + Fare

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## the response appeared on the right-hand side and was dropped

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## problem with term 6 in model.matrix: no columns are assigned

##            Df Deviance    AIC
## - Fare      1   545.53 557.53
## <none>          544.55 558.55
## + Parch     1   544.33 560.33
## + Embarked  2   544.43 562.43
## - SibSp     1   550.80 562.80
## - Age       1   555.76 567.76
## - Pclass    2   574.61 584.61
## - Sex       1   892.93 904.93
##
## Step:  AIC=557.53
## Survived ~ Pclass + Sex + Age + SibSp

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## the response appeared on the right-hand side and was dropped

## Warning in model.matrix.default(Terms, m, contrasts.arg = object$contr
asts):
## problem with term 5 in model.matrix: no columns are assigned
```

```
##             Df Deviance    AIC
## <none>           545.53 557.53
## + Fare       1   544.55 558.55
## + Parch      1   545.47 559.47
## - SibSp      1   551.14 561.14
## + Embarked   2   545.28 561.28
## - Age        1   557.52 567.52
## - Pclass     2   602.76 610.76
## - Sex        1   905.75 915.75
```

De acuerdo con los resultados del modelo_step, el mejor mejor modelo basados en el criterio AIC es el último modelo, Survived ~ Pclass + Sex + Age + SibSp, con un AIC de 557.53. Este es el modelo más eficiente con la menor complejidad, elimina variables como Fare, Parch y Embarked mientras mantiene un buen balance entre simplicidad y ajuste al conjunto de datos.

El segundo mejor modelo es Survived ~ Pclass + Sex + Age + SibSp + Fare, con un AIC de 558.55. Aunque incluye una variable adicional (Fare), su AIC ligeramente mayor sugiere que no aporta una mejora significativa al modelo.Sin embargo lo consideraremos para el resto del análisis.

Propón por lo menos los dos que consideres mejores modelos.

```
B = glm(Survived ~ Pclass + Sex + Age + SibSp, data = titanic_train, family = "binomial")
summary(B)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp, family = "binomial",
##     data = titanic_train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.107806   0.478053   8.593  < 2e-16 ***
## Pclass2     -1.295156   0.311508  -4.158 3.21e-05 ***
## Pclass3     -2.171460   0.302861  -7.170 7.51e-13 ***
## Sexmale     -3.704632   0.242002 -15.308  < 2e-16 ***
## Age         -0.030185   0.008885  -3.397 0.000681 ***
## SibSp       -0.301493   0.130504  -2.310 0.020877 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 982.80  on 730  degrees of freedom
## Residual deviance: 545.53  on 725  degrees of freedom
## AIC: 557.53
##
## Number of Fisher Scoring iterations: 5
```

Este modelo considera las variables Pclass, Sex, Age, y SibSp (número de hermanos/esposos a bordo). Todos los coeficientes son significativos, con valores de p menores a 0.05. En este modelo podemos observar lo siguiente:

- Ser hombre disminuye significativamente las probabilidades de supervivencia (coeficiente = -3.70).
- Estar en las clases 2 o 3 reduce las probabilidades de supervivencia en comparación con la clase 1, siendo el impacto más severo para la clase 3.
- La edad y el número de hermanos/esposos también tienen un impacto negativo en la probabilidad de supervivencia, aunque menor que las clases.

```
C =  glm(Survived ~ Pclass + Sex + Age + SibSp + Fare, data = titanic_tra
in, family = "binomial")
summary(C)

##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp + Fare, family = "
binomial",
##     data = titanic_train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.860003   0.540041   7.148 8.83e-13 ***
## Pclass2     -1.137654   0.350855  -3.243  0.00118 **
## Pclass3     -1.976337   0.362680  -5.449 5.06e-08 ***
## Sexmale     -3.681959   0.242967 -15.154  < 2e-16 ***
## Age         -0.029350   0.008928  -3.287  0.00101 **
## SibSp       -0.321686   0.132203  -2.433  0.01496 *
## Fare         0.002710   0.002816   0.962  0.33589
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 982.80  on 730  degrees of freedom
## Residual deviance: 544.55  on 724  degrees of freedom
## AIC: 558.55
##
## Number of Fisher Scoring iterations: 5
```

Este modelo incluye las mismas variables que el Modelo B, añadiendo la variable Fare la cuál hace referencia a la tarifa del boleto. En este modelo se hacen las siguientes observaciones:

- Los efectos de las variables Pclass, Sex, Age, y SibSp son consistentes con el Modelo B, ser hombre y/o estar en clases 2 o 3 reducen la probabilidad de supervivencia.

- La variable Fare no es significativa (p = 0.33589), indicando que, al incluir otras variables, la tarifa del boleto no aporta información adicional para predecir la supervivencia.

3. Analiza los modelos a través de:

Identificación de la Desviación residual de cada modelo

```
null_deviance_B <- B$null.deviance
null_deviance_B

## [1] 982.7966

null_deviance_C <- C$null.deviance
null_deviance_C

## [1] 982.7966
```

Identificación de la Desviación nula

```
residual_deviance_B <- B$deviance
residual_deviance_B

## [1] 545.5257

residual_deviance_C <- C$deviance
residual_deviance_C

## [1] 544.548
```

Ambos modelos parten del mismo nivel de desviación nula (982.7966). Sin embargo, el Modelo C tiene una desviación residual ligeramente menor, 544.548 contra 545.5257 del Modelo B, lo que sugiere que el Modelo C ajusta los datos marginalmente mejor.

Cálculo de la Desviación Explicada

```
explained_deviance_B <- (null_deviance_B - residual_deviance_B) / null_de
viance_B
explained_deviance_B

## [1] 0.4449251

explained_deviance_C <- (null_deviance_C - residual_deviance_C) / null_de
viance_C
explained_deviance_C

## [1] 0.4459199
```

El Modelo C explica el 44.59% de la variación en los datos, ligeramente superior al 44.49% del Modelo B. Aunque la diferencia es mínima, el Modelo C tiene una leve ventaja en términos de capacidad explicativa.

Prueba de la razón de verosimilitud

```
anova(B, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Survived
##
## Terms added sequentially (first to last)
##
##
##        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                     730     982.80
## Pclass  2    60.93       728     921.87 5.886e-14 ***
## Sex     1   361.86       727     560.01 < 2.2e-16 ***
## Age     1     8.87       726     551.14  0.002899 **
## SibSp   1     5.62       725     545.53  0.017797 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(C, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Survived
##
## Terms added sequentially (first to last)
##
##
##        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                     730     982.80
## Pclass  2    60.93       728     921.87 5.886e-14 ***
## Sex     1   361.86       727     560.01 < 2.2e-16 ***
## Age     1     8.87       726     551.14  0.002899 **
## SibSp   1     5.62       725     545.53  0.017797 *
## Fare    1     0.98       724     544.55  0.322777
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Para el Modelo B, todas las variables son significativas ($p < 0.05$), destacando que Sex y Pclass contribuyen en gran manera al ajuste. Mientras que en el Modelo C, al agregar Fare, esta variable no resulta significativa ($p = 0.322$). Esto sugiere que su inclusión no mejora el modelo.

Comparación entre los modelos B y C

```
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some
```

```
anova(B,C,test="LR")
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ Pclass + Sex + Age + SibSp
## Model 2: Survived ~ Pclass + Sex + Age + SibSp + Fare
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       725     545.53
## 2       724     544.55  1  0.97766   0.3228
```

La comparación entre el Modelo B y el Modelo C indica que agregar Fare no aporta una mejora estadísticamente significativa al modelo (p = 0.322).

Define cuál es el mejor modelo

El Modelo B es el mejor modelo porque logra un ajuste casi igual al Modelo C pero es más sencillo (menos variables), logra el equilibrio ideal entre simplicidad y precisión. Esto es importante en el contexto del problema, ya que un modelo más simple es preferible siempre que capture la relación subyacente de manera adecuada. La variable Fare, aunque conceptualmente relevante, no aporta una mejora significativa al modelo de predicción.

Escribe su ecuación, analiza sus coeficientes y detecta el efecto de cada predictor en la clasificación.

```
coefficients(B)
```

```
## (Intercept)     Pclass2     Pclass3     Sexmale         Age       SibS
p
##  4.10780618 -1.29515598 -2.17146022 -3.70463195 -0.03018527 -0.3014926
2
```

```
cat("Modelo B:\n", "Survived  = ", coef(B)[1], ... = coef(B)[2], "* Pclas
s2" ,coef(B)[3], "* Pclass3", coef(B)[4], "* Sexmale", coef(B)[5], "* Age
","  ", coef(B)[6], "* SibSp",  "\n\n")
```

```
## Modelo B:
##  Survived  =  4.107806 -1.295156 * Pclass2 -2.17146 * Pclass3 -3.70463
2 * Sexmale -0.03018527 * Age    -0.3014926 * SibSp
```

```r
coefficients(C)
```

```
##  (Intercept)      Pclass2      Pclass3      Sexmale          Age
SibSp
##  3.860003419 -1.137654498 -1.976336886 -3.681959443 -0.029350325 -0.32
1686041
##         Fare
##  0.002709997
```

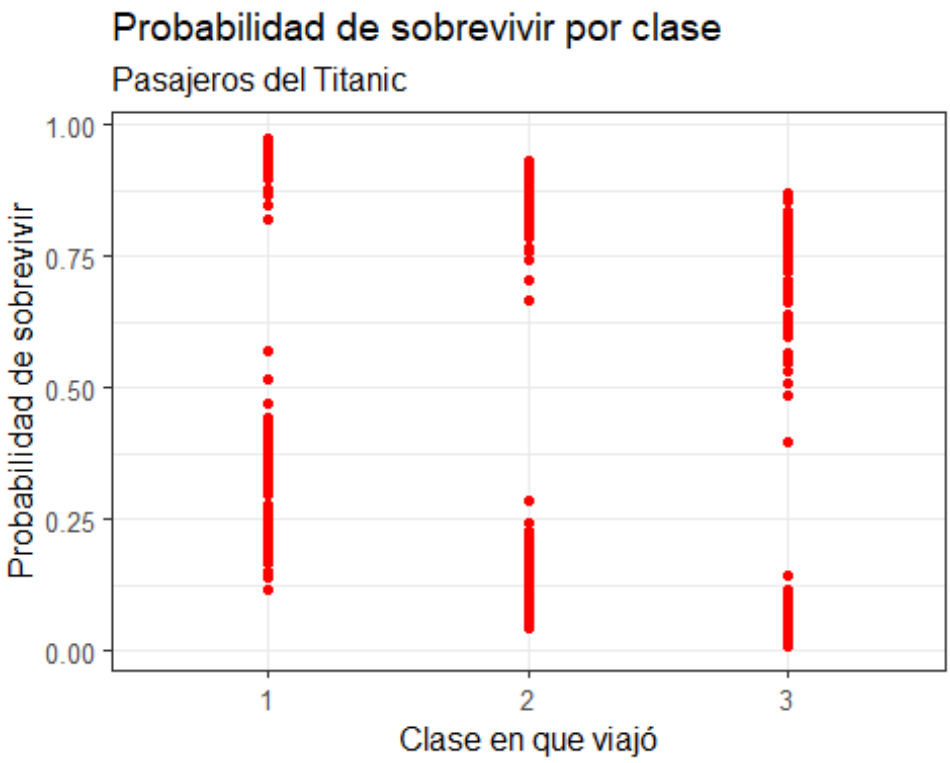
```r
cat("Modelo C:\n", "Survived  = ", coef(C)[1], ... = coef(C)[2], "* Pclas
s2" ,coef(C)[3], "* Pclass3", coef(C)[4], "* Sexmale", coef(C)[5], "* Age
","  ", coef(C)[6], "* SibSp", " + ", coef(C)[7], "Fare",  "\n\n")
```

```
## Modelo C:
##  Survived  =  3.860003 -1.137654 * Pclass2 -1.976337 * Pclass3 -3.6819
59 * Sexmale -0.02935032 * Age    -0.321686 * SibSp  +  0.002709997 Fare
```

Ambos modelos B y C buscan predecir la probabilidad de supervivencia en función de varias características. En el Modelo B, todos los coeficientes de los predictores tienen un efecto negativo sobre la supervivencia, lo que indica que pertenecer a la segunda o tercera clase, ser hombre, ser mayor en edad o tener más hermanos/esposos a bordo disminuyen la probabilidad de sobrevivir. En el Modelo C, los coeficientes son similares a los del Modelo B, aunque ligeramente menos negativos, lo que implica un efecto algo menor en la disminución de la supervivencia. Además, el Modelo C incluye el predictor Fare con un coeficiente positivo muy pequeño, sugiriendo que un incremento en la tarifa pagada está asociado con una ligera, aunque casi insignificante, mejora en la probabilidad de supervivencia. En resumen, ambos modelos indican que la clase de boleto, el género, la edad y la cantidad de familiares a bordo son factores determinantes para la supervivencia.

```r
p_pred = B$fitted.values
M_pred = data.frame(titanic_train[,c(2,3,4,5,6)],p_pred)

ggplot(M_pred, aes( x = Pclass)) +
geom_point(aes(y=M_pred$p_pred), size=1.5,color="red") +
  labs(x="Clase en que viajó", y="Probabilidad de sobrevivir",
       title="Probabilidad de sobrevivir por clase",
       subtitle="Pasajeros del Titanic",
       col="")+
  theme_bw(base_size = 12)
```

```
## Warning: Use of `M_pred$p_pred` is discouraged.
## ℹ Use `p_pred` instead.
```



El gráfico muestra cómo varía la probabilidad predicha de sobrevivir según la clase en la que viajaron los pasajeros del Titanic. En general, los pasajeros de primera clase tienen mayores probabilidades predichas de sobrevivir, seguidos por los de segunda y tercera clase. Esto sugiere que la variable Pclass es significativa, ya que existe una clara relación entre la clase del pasajero y su probabilidad de supervivencia.

4. Analisis de las predicciones para los datos de entrenamiento

Elabora la matriz de confusión

```
library(vcd)
```

```
## Loading required package: grid
```

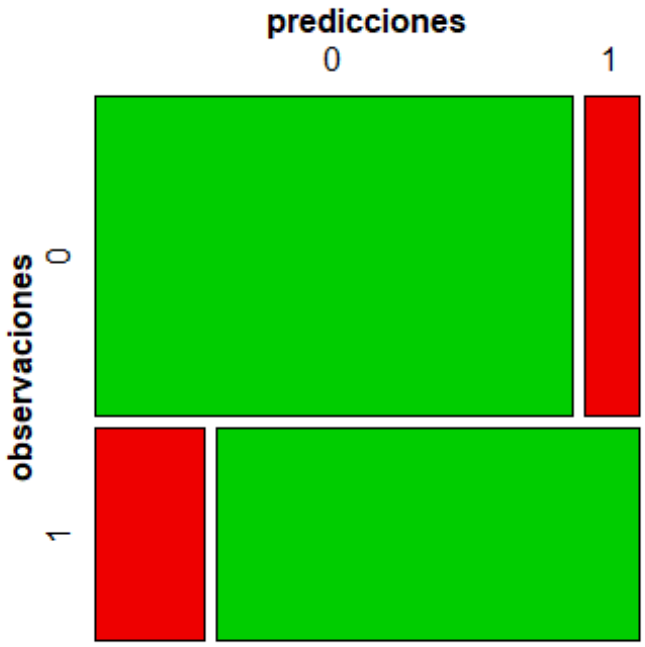
```
##
## Attaching package: 'vcd'
```

```
## The following object is masked from 'package:ISLR':
##
##     Hitters
```

```
predicciones <- ifelse(test = B$fitted.values > 0.5, yes = 1, no = 0)
M_B <- table(B$model$Survived, predicciones, dnn = c("observaciones", "pr
edicciones"))
M_B
```

```
##              predicciones
## observaciones   0   1
##             0 395  45
##             1  60 231
```

```
mosaic(M_B, shade = T, colorize = T,
       gp = gpar(fill = matrix(c("green3", "red2", "red2", "green3"), 2,
2)))
```



La matriz de confusión indica que el modelo predice correctamente 395 pasajeros que no sobrevivieron (verdaderos negativos) y 231 que sí lo hicieron (verdaderos positivos). Sin embargo, clasifica incorrectamente 45 pasajeros como sobrevivientes cuando no lo son y 60 como no sobrevivientes cuando sí lo son. Esto indica que el modelo tiene mayor dificultad para identificar correctamente a los sobrevivientes.

```
Ac = (M_B[1,1]+M_B[2,2])/sum(M_B)
cat("La Exactitud (accuracy) del modelo es", Ac,"\n")
```

```
## La Exactitud (accuracy) del modelo es 0.8563611
```

```
Se = M_B[1,1]/sum(M_B[1,])
cat("La Sensibilidad del modelo es", Se,"\n")
```

```
## La Sensibilidad del modelo es 0.8977273
```

```
Sp = M_B[2,2]/sum(M_B[2,])
cat("La Especificidad del modelo es", Sp,"\n")
```

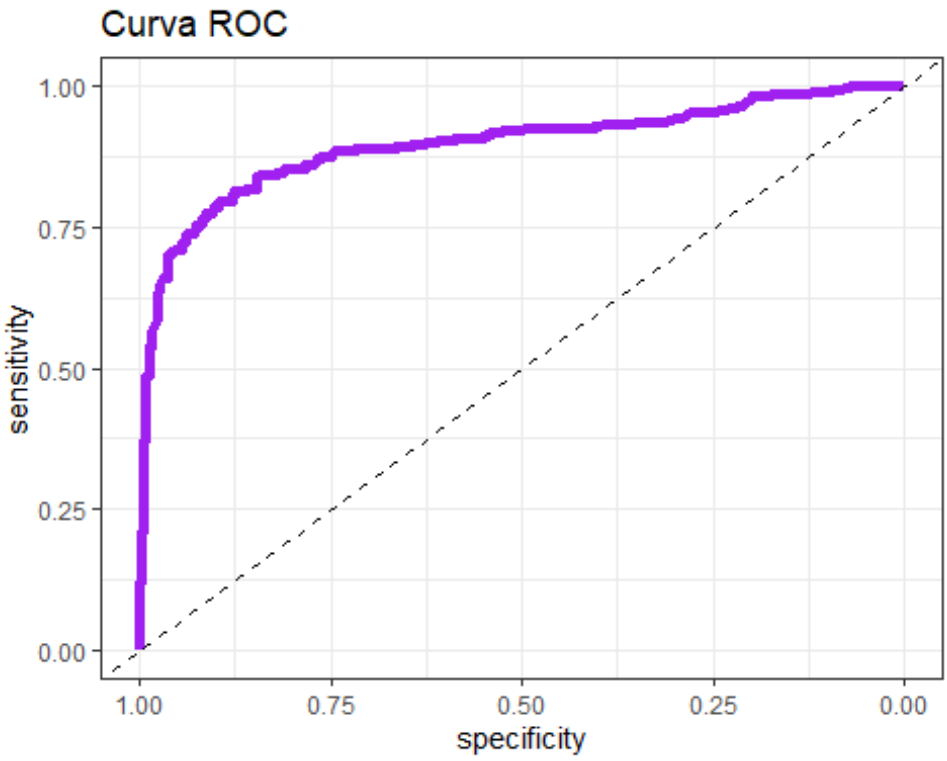
```
## La Especificidad del modelo es 0.7938144

P = M_B[1,1]/sum(M_B[,1])
cat("La Precisión del modelo es", P,"\n")

## La Precisión del modelo es 0.8681319
```

Con una exactitud de 85.63%, el modelo es relativamente bueno para predecir los resultados, la sensibilidad (89.77%) indica que el modelo es eficaz para detectar a los sobrevivientes, la especificidad (79.38%) muestra una menor capacidad para identificar correctamente a los no sobrevivientes. La precisión (86.81%) refuerza la idea de que el modelo tiene un buen balance general.

Elabora la Curva ROC

Para hacer la curva, es necesario crear las predicciones para el data set de entrenamiento. El comando *roc* cálculará la sensibilidad y la especificidad para los datos obtenidos.

```
pred = predict(B, data = titanic_train, type = 'response')

library(pROC)
ROC <- roc(response=titanic_train$Survived, predictor=pred)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

ROC

##
## Call:
## roc.default(response = titanic_train$Survived, predictor = pred)
##
## Data: pred in 440 controls (titanic_train$Survived 0) < 291 cases (tit
anic_train$Survived 1).
## Area under the curve: 0.8963

ggroc(ROC, color = "purple", size = 2) + geom_abline(slope = 1, intercept
= 1, linetype ='dashed') + labs(title = "Curva ROC") + theme_bw()
```

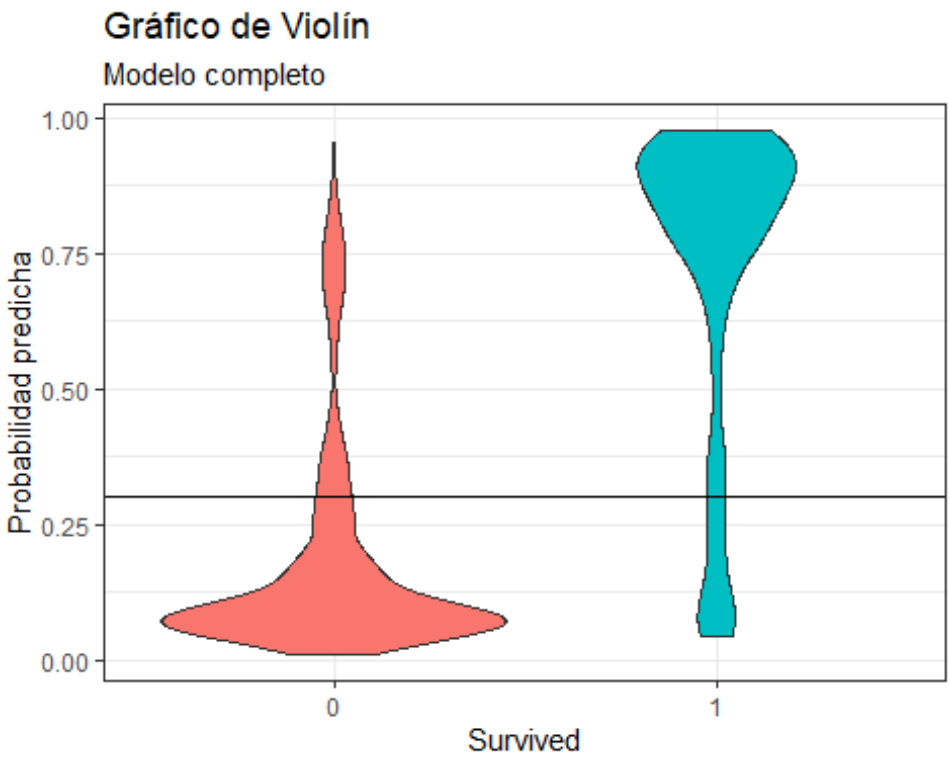


La curva ROC muestra una buena separación entre las clases predichas, con un área bajo la curva (AUC) de 0.8963, lo que indica un excelente desempeño del modelo. Este valor sugiere que, en un 89.63% de los casos, el modelo puede diferenciar correctamente entre un sobreviviente y un no sobreviviente.

Elabora el gráfico de violín

```
v_d = data.frame(Survived=titanic_train$Survived,pred=pred)

ggplot(data=v_d, aes(x=Survived, y=pred, group=Survived, fill=factor(Surv
ived))) +
  geom_violin() + geom_abline(aes(intercept=0.3,slope=0))+
  theme_bw() +
  guides(fill=FALSE) +
  labs(title='Gráfico de Violín', subtitle='Modelo completo', y='Probabil
idad predicha')
```

```
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "
none" instead as
## of ggplot2 3.3.4.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning
was
## generated.
```



El gráfico de violín muestra la distribución de las probabilidades predichas para los sobrevivientes (1) y los no sobrevivientes (0). Las probabilidades están claramente separadas, lo que confirma que el modelo puede distinguir entre ambas clases.

Concluye sobre el modelo basándote en las predicciones de los datos de entrenamiento.

El modelo tiene un buen desempeño general, con una alta exactitud y AUC. Detecta bien a los sobrevivientes y ofrece predicciones diferenciadas por clase, como lo muestran las distribuciones. Sin embargo, tiene limitaciones en la identificación precisa de los sobrevivientes. Esto podría mejorarse ajustando el umbral de decisión o considerando variables adicionales. En general, el modelo es confiable para interpretar tendencias y clasificar a los pasajeros según su probabilidad de supervivencia.

5. Validación del modelo con la base de datos de validación

Elije un umbral de clasificación óptimo

Elección del umbral de clasificación (punto de corte)

Se trabaja con la base de datos de validación (*titanic_valid*) y se realiza el gráfico de la Exactitud, Sensibilidad, Especificidad y Precisión para distintos valores del umbral de clasificación. Se siguen los siguientes pasos:

1. Predicción en los datos de validación con el modelo elegido
2. Se definen los umbrales de clasificación: irán desde 0.05 hasta 0.95.
3. Se definen las métricas de la matriz de confusión para cada umbral de clasificación
4. Se prepara el conjunto de datos: se quitan los NA y se agrega la columna de umbrales de clasificaición
5. Se le da un formato a la base de datos para que pueda ser graficada más fácilmente.

Generación de base de datos para graficar

```
pred_val = predict(B, newdata=titanic_valid, type='response')
clase_real = titanic_valid$Survived

datosV = data.frame(accuracy=NA, recall=NA, specificity = NA, precision=N
A)

for (i in 5:95){
  clase_predicha = ifelse(pred_val>i/100,1,0)

##Creamos la matriz de confusión
cm= table(clase_predicha,clase_real)

## AccurAcy: Proporción de correctamente predichos
datosV[i,1] = (cm[1,1]+cm[2,2])/(cm[1,1]+cm[1,2]+cm[2,1]+cm[2,2])
## Recall: Tasa de positivos correctamente predichos
datosV[i,2] = (cm[2,2])/(cm[1,2]+cm[2,2])
## Specificity: Tasa de negativos correctamente predichos
datosV[i,3] = cm[1,1]/(cm[1,1]+cm[2,1])
## Precision: Tasa de bien clasificados entre los clasificados como posit
ivos
datosV[i,4] = cm[2,2]/(cm[2,1]+cm[2,2])
}

## Se limpia el conjunto de datos
datosV = na.omit(datosV)
datosV$umbral = seq(0.05,0.95,0.01)
```

Formato de datos

- Se crea la variable *métrica* que será una variable categórica para las métricas (Exactitud, Sensibilidad, Especificidad y Precisión)
- Los valores de las métricas se ponen en una sola columna.

- Se identifican las métricas para los distintos umbrales con la variable 'umbral'.

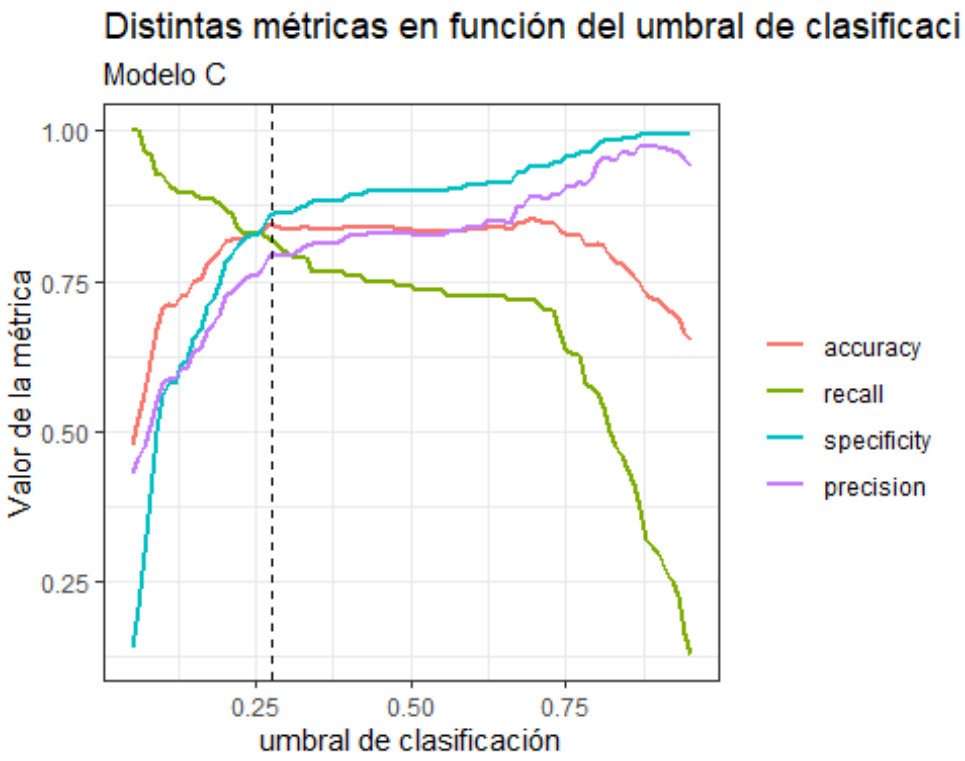
```
library(reshape2)

##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths

datosV_m <- reshape2::melt(datosV,id.vars=c('umbral'))
colnames(datosV_m)[2] <- c('Metrica')
```

Gráfica

En la gráfica se define cuál es el mejor umbral de clasificación dependiendo de cuál métrica es más importante en el contexto del problema (Exactitud, Sensibilidad, Especificidad o Precisión). Si no hay una métrica de preferencia, se opta por escoger el máximo valor de que pueden tener estás métricas en conjunto. En cualquier caso da valores a u para mover el umbral de clasificación y observar como se comporta con respecto a las métricas.

```
library(ggplot2)

u = 0.275

ggplot(data=datosV_m, aes(x=umbral,y=value,color=Metrica)) + geom_line(si
ze=1) + theme_bw() +
  labs(title= 'Distintas métricas en función del umbral de clasificación'
,
       subtitle= 'Modelo C',
       color="", x = 'umbral de clasificación', y = 'Valor de la métrica'
) +
  geom_vline(xintercept=u, linetype="dashed", color = "black")

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.
4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning
was
## generated.
```



La gráfica muestra cómo varían las métricas del modelo (Exactitud, Sensibilidad, Especificidad y Precisión) en función del umbral de clasificación. A medida que el umbral aumenta, las métricas se comportan de manera diferente: la sensibilidad (recall) disminuye rápidamente, ya que un umbral más alto hace que el modelo clasifique menos datos como positivos. Por otro lado, la especificidad aumenta, lo que indica que el modelo clasifica mejor a los negativos. La exactitud (accuracy) y la precisión (precision) tienen tendencias más estables y alcanzan picos en valores intermedios del umbral.

El mejor umbral depende del objetivo del modelo. Si la prioridad es maximizar la sensibilidad para capturar el mayor número de positivos (por ejemplo, si los sobrevivientes son críticos en el contexto del problema), un umbral bajo como 0.2 es adecuado. Sin embargo, si se busca un balance entre exactitud, sensibilidad y especificidad, el gráfico sugiere que el umbral óptimo está alrededor de 0.25-0.30, donde estas métricas tienen valores altos y equilibrados.

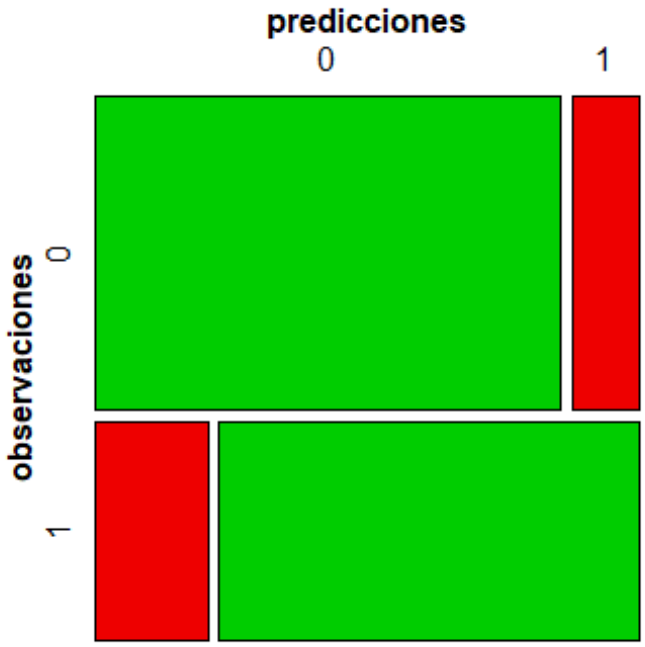
En conclusión, el mejor umbral dependerá de la métrica prioritaria, pero un rango entre 0.25 y 0.30 parece ser un un buen umbral para mantener un buen desempeño general del modelo, yo he decidido tomar 0.275

Elabora la matriz de confusión con el umbral de clasificación óptimo

```
prediccionesV = ifelse(pred_val > 0.275, yes = 1, no = 0)
M_Cv <- table(prediccionesV, titanic_valid$Survived, dnn = c("observacion
es", "predicciones"))
M_Cv
```

```
##              predicciones
## observaciones   0   1
##             0 161  23
##             1  27 101
```

```
mosaic(M_Cv, shade = T, colorize = T, gp = gpar(fill = matrix(c("green3",
"red2", "red2", "green3"), 2, 2)))
```



La matriz de confusión para la base de validación muestra que el modelo predijo correctamente 161 pasajeros que no sobrevivieron (verdaderos negativos) y 101 que sí sobrevivieron (verdaderos positivos). Sin embargo, clasificó erróneamente a 23 pasajeros como sobrevivientes cuando no lo fueron y a 27 como no sobrevivientes cuando sí lo fueron.

```
AcV = (M_Cv[1,1]+M_Cv[2,2])/sum(M_Cv)
cat("La Exactitud (accuracy) del modelo es", AcV,"\n")
```

```
## La Exactitud (accuracy) del modelo es 0.8397436
```

```
SeV = M_Cv[1,1]/sum(M_Cv[1,])
cat("La Sensibilidad del modelo es", SeV,"\n")
```

```
## La Sensibilidad del modelo es 0.875
```

```
SpV = M_Cv[2,2]/sum(M_Cv[2,])
cat("La Especificidad del modelo es", SpV,"\n")
```

```
## La Especificidad del modelo es 0.7890625
```

```r
PV = M_Cv[1,1]/sum(M_Cv[,1])
cat("La Precisión del modelo es", PV,"\n")

## La Precisión del modelo es 0.856383
```

Las métricas muestran un buen desempeño general del modelo la Exactitud (Accuracy) del 83.97% indica que el modelo clasifica correctamente a la mayoría de los pasajeros, la sensibilidad (Recall) de 87.5% implica que el modelo identifica la mayoría de los sobrevivientes correctamente.Especificidad con 78.91% muestra que también clasifica bien a los no sobrevivientes, aunque con menor precisión que a los sobrevivientes y finalmente la precisión con un valor de 85.64% nos indica que la mayoría de los pasajeros clasificados como sobrevivientes realmente lo son. Estas métricas confirman que el umbral elegido de 0.275 logra un balance adecuado entre la sensibilidad y la especificidad.

6. Elabora el testeo con la base de datos de prueba.

```r
titanic_test <- titanic_test %>%
  mutate(Sex = as.factor(Sex),
         Embarked = as.factor(Embarked),
         Pclass = as.factor(Pclass))

test_predictions <- predict(B, newdata = titanic_test, type = "response")
test_pred_class <- ifelse(test_predictions > 0.275, 1, 0)

summary(B)

##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp, family = "binomia
l",
##     data = titanic_train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.107806   0.478053   8.593  < 2e-16 ***
## Pclass2     -1.295156   0.311508  -4.158 3.21e-05 ***
## Pclass3     -2.171460   0.302861  -7.170 7.51e-13 ***
## Sexmale     -3.704632   0.242002 -15.308  < 2e-16 ***
## Age         -0.030185   0.008885  -3.397 0.000681 ***
## SibSp       -0.301493   0.130504  -2.310 0.020877 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 982.80  on 730  degrees of freedom
## Residual deviance: 545.53  on 725  degrees of freedom
## AIC: 557.53
```

```
##
## Number of Fisher Scoring iterations: 5
```

El modelo logístico incluye como variables predictoras significativas:

- Pclass: Los pasajeros de segunda y tercera clase tienen menor probabilidad de sobrevivir en comparación con los de primera clase con los sguientes valores, Pclass2 = -1.29 y Pclass3 = -2.17.
- Sexo: Ser hombre disminuye drásticamente la probabilidad de sobrevivir (-3.70), lo que es consistente con los datos históricos del Titanic.
- Edad: A mayor edad, la probabilidad de sobrevivir disminuye ligeramente (-0.03).
- SibSp: Tener más familiares a bordo también reduce la probabilidad de sobrevivir (SibSp: -0.30).

El modelo es estadísticamente significativo en todas las variables (valores p menores a 0.05). Además, el AIC del modelo (557.53) y la reducción en la devianza residual frente a la nula indican que el modelo tiene un buen ajuste general.

7. Concluye en el contexto del problema:

Define las principales características que influyen en el modelo seleccionado e interpretalas: ¿qué características tuvieron las personas que sobrevivieron?

De acuerdo con los resultados del modelo, las principales características que determinaron la probabilidad de supervivencia de los pasajeros son Pclass, Sex, Age y SibSp. Los pasajeros de primera clase tenían mayores probabilidades de sobrevivir, mientras que los de segunda y tercera clase enfrentaron un riesgo significativamente mayor debido a desigualdades al acceso de los recursos como los botes salvavidas y las condiciones de evacuación. Ser mujer aumentó notablemente la probabilidad de supervivencia en comparación con los hombres, reflejando lo que todos hemos escuchado de este evento histórico de “mujeres y niños primero” durante la evacuación. Las personas de mayor edad tenían una ligera desventaja en la probabilidad de sobrevivir, probablemente debido a limitaciones físicas y/o prioridades en el rescate para pasajeros más jóvenes. Además, a mayor cantidad de familiares a bordo (hermanos/esposos), menor era la probabilidad de supervivencia, lo cual podría deberse a dificultades en la coordinación durante la evacuación o la necesidad de priorizar a ciertos miembros del grupo familiar o decisión de permanecer todos juntos.

Interpreta los coeficientes del modelo

- Intercepto: Representa la probabilidad base de sobrevivir para un pasajero de referencia (de primera clase, mujer, edad cero y sin familiares), siendo positiva y significativa.

- Pclass2 y Pclass3: Ambos tienen coeficientes negativos indicando que viajar en segunda o tercera clase reduce la probabilidad de supervivencia en comparación con primera clase.
- Sexmale: Este fuerte coeficiente negativo nos dice que ser hombre disminuye drásticamente la probabilidad de sobrevivir. Siendo esto lo que más reduce la probabilidad de supervivencia.
- Age: Cada año adicional reduce ligeramente la probabilidad de sobrevivir, aunque su efecto es menor en comparación con otras variables.
- SibSp: Por cada familiar adicional a bordo, la probabilidad de supervivencia disminuye moderadamente.

Define cuál es el mejor umbral de clasificación y por qué

El mejor umbral de clasificación identificado fue 0.275. Este valor fue seleccionado porque proporciona un balance adecuado entre las métricas clave del modelo obteniendo alta sensibilidad (se logra identificar a la mayoría de los sobrevivientes, lo que es crucial en el contexto del Titanic donde salvar vidas es la priorida), aceptable especificidad (aunque menor que la sensibilidad, permite reducir errores al clasificar correctamente a los no sobrevivientes), exactitud global y precisión, estos últimos muestran que el modelo clasifica correctamente a la mayoría de los pasajeros, con una tasa alta de acierto entre los clasificados como sobrevivientes.

En resumen, el umbral de 0.275 logra maximizar el balance entre salvar el mayor número de sobrevivientes y minimizar los errores en la clasificación, siendo el más adecuado para este problema.