Actividad Integradora 2 Titanic Fer

Fernanda Pérez

2024-11-19

Actividad Integradora 2

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.

```
librerias = c('tidyverse','broom','ISLR','GGally','modelr','cowplot','rlang','modelr','tibble','Metrics

for (lib in librerias) {
  if (!require(lib, character.only = TRUE)) {
    install.packages(lib, dependencies = TRUE)
    library(lib, character.only = TRUE)
  }
}
```

```
## Cargando paquete requerido: tidyverse

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Cargando paquete requerido: broom
##
## Cargando paquete requerido: ISLR
##
## Cargando paquete requerido: GGally

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

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
## Cargando paquete requerido: modelr
##
## Adjuntando el paquete: 'modelr'
```

```
##
## The following object is masked from 'package:broom':
##
##     bootstrap
##
## Cargando paquete requerido: cowplot
##
## Adjuntando el paquete: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
##
## Cargando paquete requerido: rlang
##
## Adjuntando el paquete: 'rlang'
##
## The following objects are masked from 'package:purrr':
##
##     %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
##     flatten_raw, invoke, splice
##
## Cargando paquete requerido: Metrics

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

##
## Adjuntando el paquete: 'Metrics'
##
## The following object is masked from 'package:rlang':
##
##     ll
##
## The following objects are masked from 'package:modelr':
##
##     mae, mape, mse, rmse
##
## Cargando paquete requerido: mice

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

##
## Adjuntando el paquete: 'mice'
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following objects are masked from 'package:base':
##
##     cbind, rbind
##
## Cargando paquete requerido: visdat
```

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

## Cargando paquete requerido: caret

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

## Cargando paquete requerido: lattice
##
## Adjuntando el paquete: 'caret'
##
## The following objects are masked from 'package:Metrics':
##
##     precision, recall
##
## The following object is masked from 'package:purrr':
##
##     lift
```

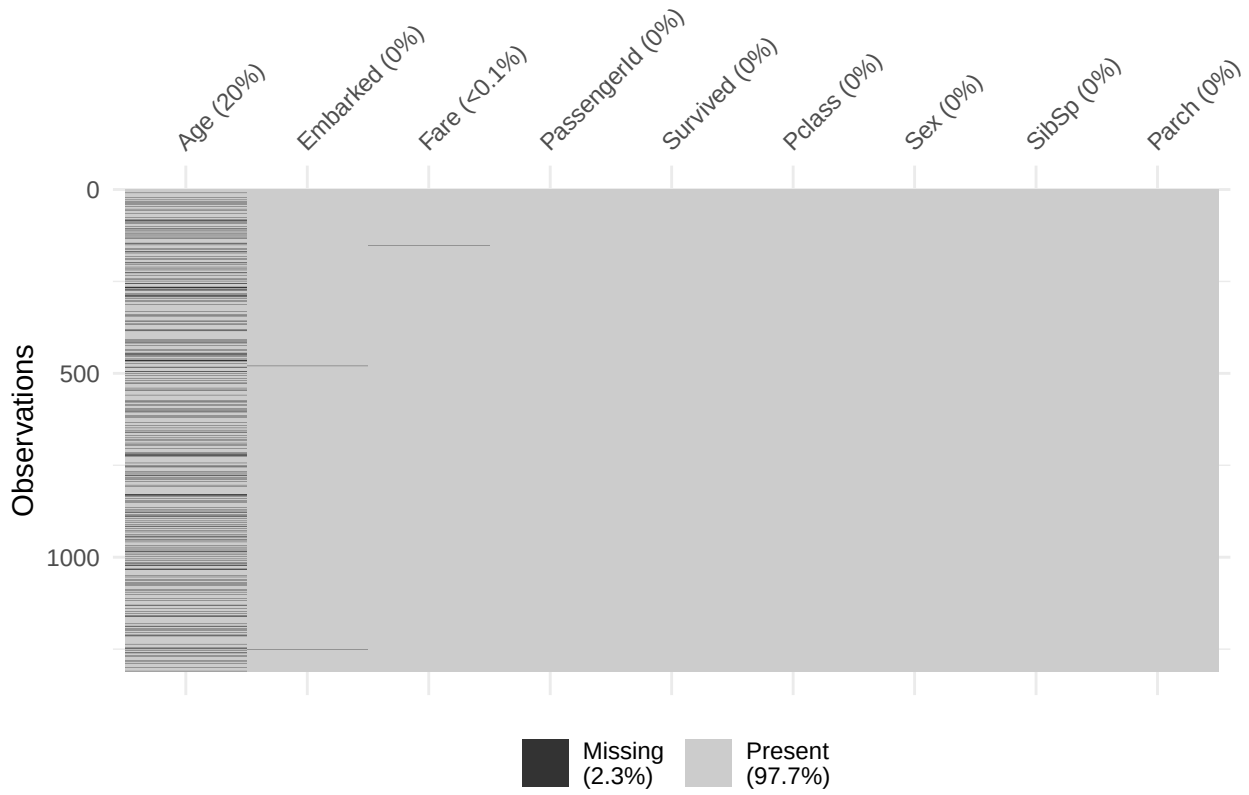
1.Prepara la base de datos Titanic:

1.a) Analiza los datos faltantes

```
M <- read.csv("D:/Downloads/Titanic.csv")
M_test <- read.csv("D:/Downloads/Titanic_test.csv")

M1 <- M[, -c(4, 9, 11)]

categoricas <- c("Survived", "Pclass", "Sex", "Embarked")
for (var in categoricas) {
  M1[, var] <- as.factor(M1[, var])
}

str(M1)
```

```
## 'data.frame':    1309 obs. of  9 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Survived   : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 2 1 2 1 ...
##  $ Pclass     : Factor w/ 3 levels "1","2","3": 3 3 2 3 3 3 3 2 3 3 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 1 2 1 2 ...
##  $ Age        : num  34.5 47 62 27 22 14 30 26 18 21 ...
##  $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
##  $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1 3 ...
```

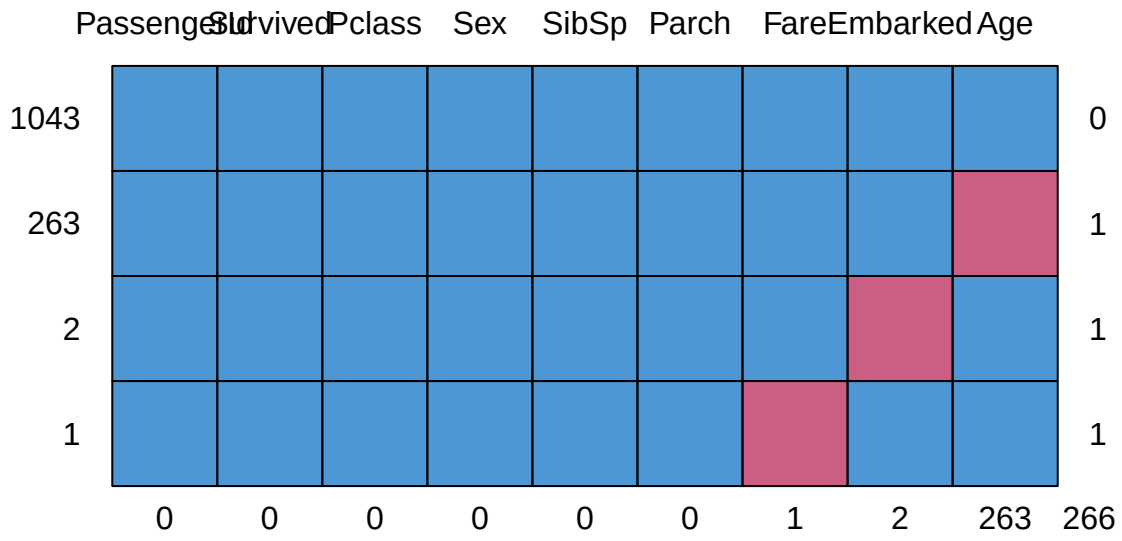
```
library(visdat)
vis_miss(M1, sort_miss = TRUE)
```




```r
N <- apply(is.na(M1), MARGIN = 2, FUN = sum)
P <- round(100 * N / nrow(M1), 2)
NP <- data.frame(Número = N, Porcentaje = P)
row.names(NP) <- names(M1)
print(NP)
```

```
##             Número Porcentaje
## PassengerId      0       0.00
## Survived         0       0.00
## Pclass           0       0.00
## Sex              0       0.00
## Age            263      20.09
## SibSp            0       0.00
## Parch            0       0.00
## Fare             1       0.08
## Embarked         2       0.15
```

```r
library(mice)
md.pattern(M1)
```




```
##      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
```

Este análisis de datos faltantes en el dataset (Titanic) ajustado nos dice que la mayoría de las variables están completas, contamos con un 97.7% de los datos y solo tenemos missing un (2.3%). Pero hay valores faltantes en tres variables: Age (20.09%), Fare (0.08%), y Embarked (0.15%), vemos que de las tres la variable Age es la que cuenta con el mayor porcentaje de datos faltantes, lo que podría alterar significativamente los resultados del modelo, ya que la edad es un factor que podria ser relevante para la supervivencia dado la politica de niños y mujere primero. Podemos deducir que el patrón de datos faltantes indica que estas ausencias son aleatorias y no están correlacionadas con otras variables.

1.b) Realiza un análisis descriptivo

```r
# variables continuas
summary(M1[, c("Age", "Fare")])
```

```
##       Age             Fare
##  Min.   : 0.17   Min.   :  0.000
##  1st Qu.:21.00   1st Qu.:  7.896
```

```
##  Median :28.00   Median : 14.454
##  Mean   :29.88   Mean   : 33.295
##  3rd Qu.:39.00   3rd Qu.: 31.275
##  Max.   :80.00   Max.   :512.329
##  NA's   :263     NA's   :1
```

```
tapply(as.numeric(M1$Survived) - 1, M1$Pclass, mean)
```

```
##         1         2         3
## 0.5758514 0.4223827 0.2693935
```

```
library(ggplot2)
ggplot(M1, aes(x = Pclass, fill = Survived)) +
  geom_bar(position = "fill") +
  labs(title = "Proporción de sobrevivientes por clase",
       x = "Clase",
       y = "Proporción",
       fill = "Sobrevivió") +
  theme_minimal()
```

Vemos que la distribución de la variable Age muestra un rango amplio de edades (0.17 a 80 años), tiene una mediana de 28 años, indicando que la mayoría de los pasajeros eran adultos jóvenes. Y en Fare, existe una gran dispersión con un valor máximo de 512.33, sugiriendo posibles valores atípicos. En el gráfico vemos la proporción de sobrevivientes por clase, vemos que la probabilidad de supervivencia disminuye significativamente al pasar de la primera a la tercera clase, lo que destaca la importancia del estatus socioeconómico como factor determinante en la supervivencia en este evento.

```r
prop.table(table(M1$Survived))
```

```
##
##         0         1
## 0.6226127 0.3773873
```

```r
prop.table(table(M1$Pclass))
```

```
##
##         1         2         3
## 0.2467532 0.2116119 0.5416348
```

```r
prop.table(table(M1$Sex))
```

```
##
##    female      male
## 0.3559969 0.6440031
```

```r
prop.table(table(M1$Embarked))
```

```
##
##          C          Q          S
## 0.20657995 0.09410865 0.69931140
```

Vemos que el 62.26% de los pasajeros no sobrevivió, la mayoría hombres (64.40%) y de tercera clase (54.16%). La mayoría embarcó en Southampton (69.93%), lo que podría influir en las tasas de supervivencia que vemos.

1.c) 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.

```r
library(caret)
set.seed(123)
M_indice <- createDataPartition(M1$Survived, p = 0.7, list = FALSE, times = 1)

M_train <- M1[M_indice, ]
M_valid <- M1[-M_indice, ]

prop.table(table(M_train$Survived))
```

```
##
##         0         1
## 0.6226827 0.3773173
```

```r
prop.table(table(M_valid$Survived))
```

```
##
##        0        1
## 0.622449 0.377551
```

```r
prop.table(table(M1$Survived))
```

```
##
##         0         1
## 0.6226127 0.3773873
```

Partimos los datos en 70-30 conservando la proporción de supervivientes de la base original (37.73% sobrevivientes y 62.27% no sobrevivientes) en los conjuntos de entrenamiento y validación. Con esto se garantiza que ambos conjuntos representen adecuadamente la distribución original, evitando sesgos en el modelo.

2.Con la base de datos de entrenamiento, encuentra un modelo logístico para encontrar el mejor conjunto de predictores que auxilien a clasificar la dirección de cada observación.

Auxiliate del criterio de AIC para determinar cuál es el mejor modelo. Propón por lo menos los dos que consideres mejores modelos.

```r
M_train_clean <- na.omit(M_train)

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

# Selección de modelo usando el criterio AIC
step_model <- step(A, direction = "both", trace = 1)
```

```
## Start:  AIC=584.63
## Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked
##
##            Df Deviance    AIC
## - Embarked  2   564.81 580.81
## - Parch     1   565.19 583.19
## - Fare      1   566.08 584.08
## <none>          564.63 584.63
## - SibSp     1   572.74 590.74
## - Age       1   581.85 599.85
## - Pclass    2   592.78 608.78
## - Sex       1   886.23 904.23
##
## Step:  AIC=580.81
## Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare
##
##            Df Deviance    AIC
## - Parch     1   565.44 579.44
## - Fare      1   566.49 580.49
## <none>          564.81 580.81
## + Embarked  2   564.63 584.63
## - SibSp     1   573.25 587.25
```

```
## - Age       1   582.23 596.23
## - Pclass    2   594.06 606.06
## - Sex       1   896.08 910.08
##
## Step:  AIC=579.44
## Survived ~ Pclass + Sex + Age + SibSp + Fare
##
##            Df Deviance    AIC
## - Fare      1   566.76 578.76
## <none>          565.44 579.44
## + Parch     1   564.81 580.81
## + Embarked  2   565.19 583.19
## - SibSp     1   575.97 587.97
## - Age       1   582.61 594.61
## - Pclass    2   596.28 606.28
## - Sex       1   901.31 913.31
##
## Step:  AIC=578.76
## Survived ~ Pclass + Sex + Age + SibSp
##
##            Df Deviance    AIC
## <none>          566.76 578.76
## + Fare      1   565.44 579.44
## + Parch     1   566.49 580.49
## + Embarked  2   566.31 582.31
## - SibSp     1   576.39 586.39
## - Age       1   584.50 594.50
## - Pclass    2   619.73 627.73
## - Sex       1   908.36 918.36
```

```r
# Modelo B
B <- glm(Survived ~ Pclass + Sex + Age + Fare + Embarked,
         family = "binomial", data = M_train_clean)
summary(B)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + Fare + Embarked,
##     family = "binomial", data = M_train_clean)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.591862   0.517946   6.935 4.07e-12 ***
## Pclass2     -0.897268   0.345937  -2.594  0.00949 **
## Pclass3     -1.876690   0.359151  -5.225 1.74e-07 ***
## Sexmale     -3.415794   0.226632 -15.072  < 2e-16 ***
## Age         -0.029130   0.008407  -3.465  0.00053 ***
## Fare         0.001138   0.002164   0.526  0.59890
## EmbarkedQ    0.115088   0.557652   0.206  0.83649
## EmbarkedS   -0.205927   0.283299  -0.727  0.46729
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 989.11  on 738  degrees of freedom
## Residual deviance: 575.13  on 731  degrees of freedom
## AIC: 591.13
##
## Number of Fisher Scoring iterations: 5
```

```r
# Modelo C
C <- glm(Survived ~ Pclass + Sex + Age + Fare,
         family = "binomial", data = M_train_clean)
summary(C)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + Fare, family = "binomial",
##     data = M_train_clean)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.481381   0.487994   7.134 9.75e-13 ***
## Pclass2     -0.961796   0.335494  -2.867 0.004146 **
## Pclass3     -1.901157   0.351147  -5.414 6.16e-08 ***
## Sexmale     -3.436005   0.225884 -15.211  < 2e-16 ***
## Age         -0.029225   0.008420  -3.471 0.000519 ***
## Fare         0.001380   0.002133   0.647 0.517662
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 989.11  on 738  degrees of freedom
## Residual deviance: 575.97  on 733  degrees of freedom
## AIC: 587.97
##
## Number of Fisher Scoring iterations: 5
```

Lo que se hizo para definir los modelos fue: El Modelo A incluye todas las variables iniciales para explicar la supervivencia. El Modelo B quita variables menos significativas según el criterio AIC, manteniendo : Pclass, Sex, Age, Fare y Embarked, y el ultimo modelo, el Modelo C, recomendado por step, optimiza aún más el ajuste al eliminar Embarked, dejando las variables más relevantes. Los dos modelos simplificados reducen el AIC y la desviación residual.

Interpretación: Con los modelos podemos identificar que las variables: `Pclass`, `Sex`, `Age` y, en menor medida, `Fare` son predictores significativos de la supervivencia. El modelo c presenta un AIC de 587.97 con una desviación residual de 575.97 e indica que pertenecer a una clase económica más baja (`Pclass3`), ser hombre (`Sexmale`) y tener mayor edad reducen significativamente la probabilidad de supervivencia del pasajero, lo cual tiene sentido por la politica del barco de niños y mujeres primeor. La inclusión de `Fare` no resulta significativa en este modelo.

---

3. Analiza los modelos a través de:

Identificación de la Desviación residual de cada modelo

```
A$deviance
```

```
## [1] 564.6328
```

```
B$deviance
```

```
## [1] 575.1259
```

```
C$deviance
```

```
## [1] 575.9697
```

El modelo A tiene la menor desviación residual (564.63), lo que indica un mejor ajuste comparado con los modelos B (575.13) y C (575.97). Esto sugiere que el modelo A, al incluir todas las variables, explica mejor los datos aunque puede ser menos simple.

Identificación de la Desviación nula

```
A$null.deviance
```

```
## [1] 989.1129
```

```
B$null.deviance
```

```
## [1] 989.1129
```

```
C$null.deviance
```

```
## [1] 989.1129
```

Vemos que la desviación nula es la misma para todos los modelos (989.11), indicandonos que la variabilidad total de los datos es la misma antes de incluir las variables predictoras. Esto es consistente, ya que todos los modelos utilizan el mismo conjunto de datos inicial.

Cálculo de la Desviación Explicada

```
pseudo_r2_A <- 1 - (A$deviance / A$null.deviance)
pseudo_r2_B <- 1 - (B$deviance / B$null.deviance)
pseudo_r2_C <- 1 - (C$deviance / C$null.deviance)

pseudo_r2_A
```

```
## [1] 0.4291523
```

```r
pseudo_r2_B
```

```
## [1] 0.4185437
```

```r
pseudo_r2_C
```

```
## [1] 0.4176906
```

El modelo A logra explicar el 42.92% de la variabilidad (pseudo R^2), indiando que es el más efectivo, seguido por el modelo B (41.85%) y el modelo C (41.76%). Indicandonos que el modelo A tiene un mejor desempeño en términos de explicación de los datos.

Prueba de la razón de verosimilitud

```r
diferencia_A <- A$null.deviance - A$deviance
gl_A <- A$df.null - A$df.residual
pchisq(diferencia_A, gl_A, lower.tail = FALSE)
```

```
## [1] 8.142708e-86
```

```r
diferencia_B <- B$null.deviance - B$deviance
gl_B <- B$df.null - B$df.residual
pchisq(diferencia_B, gl_B, lower.tail = FALSE)
```

```
## [1] 2.384777e-85
```

```r
diferencia_C <- C$null.deviance - C$deviance
gl_C <- C$df.null - C$df.residual
pchisq(diferencia_C, gl_C, lower.tail = FALSE)
```

```
## [1] 4.357386e-87
```

Para los tres modelos como los valores p son extremadamente pequeños < 0.001, esto indica que todos explican significativamente mejor la variable respuesta que el modelo nulo, confirmando la relevancia de las variables predictoras incluidas.

Define cuál es el mejor modelo

```r
anova(A, B, test = "LR")
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked
## Model 2: Survived ~ Pclass + Sex + Age + Fare + Embarked
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       729     564.63
## 2       731     575.13 -2  -10.493 0.005266 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ Pclass + Sex + Age + Fare + Embarked
## Model 2: Survived ~ Pclass + Sex + Age + Fare
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       731     575.13
## 2       733     575.97 -2 -0.84379   0.6558
```

En el análisis vemos que el modelo A es significativamente mejor que el modelo B (p = 0.005), pero no vemos una diferencia significativa entre los modelos B y C (p = 0.656). Esto nos sugiere que el modelo A, apesar de ser más complejo nos ofrece el mejor ajuste.

Despues de revisar todo el análisis realizado podemos decir que el modelo A se considera el mejor, ya que:

1. tuvo el mejor desempeño: Tiene la menor desviación residual (564.63) y el mayor pseudo R^2 (42.92%), indicando que explica mejor la variabilidad en los datos.
2. tuvo mejor razón de verosimilitud**: Es significativamente mejor que el modelo B (p = 0.005), demostrando que las variables adicionales incluidas en el modelo A aportan información relevante.
3. AIC: A pesar de quee el AIC del modelo A es un poco mayorr que el de los otros modelos, la diferencia no es suficiente para descartar su mejor capacidad explicativa.

El modelo A que incluye todas las variables predictoras :`Pclass`, `Sex`, `Age`, `SibSp`, `Parch`, `Fare`, y `Embarked`,podriamos decir que es el que tuvo mejor desempeño y aunque es más complejo, ofrece el ajuste más completo y preciso para explicar la supervivencia. Sin embargo, si buscamos simplicidad y eficiencia podría considerarse el modelo B como una alternativa razonable.

Escribe su ecuación

```r
coefficients_A <- round(A$coefficients, 3)

equation <- paste0(
  "logit(Survived) = ", coefficients_A[1], " + ",
  coefficients_A[2], " * Pclass2 + ",
  coefficients_A[3], " * Pclass3 + ",
  coefficients_A[4], " * Sexmale + ",
  coefficients_A[5], " * Age + ",
  coefficients_A[6], " * SibSp + ",
  coefficients_A[7], " * Parch + ",
  coefficients_A[8], " * Fare + ",
  coefficients_A[9], " * EmbarkedQ + ",
  coefficients_A[10], " * EmbarkedS"
)
equation
```

```
## [1] "logit(Survived) = 4.007 + -0.953 * Pclass2 + -1.852 * Pclass3 + -3.577 * Sexmale + -0.036 * Age
```

analiza sus coeficientes

```r
summary_A <- summary(A)
summary_A$coefficients
```

```
##                 Estimate  Std. Error      z value     Pr(>|z|)
## (Intercept)  4.007401104 0.548873351   7.30113986 2.853399e-13
## Pclass2     -0.953392336 0.352452620  -2.70502269 6.829972e-03
## Pclass3     -1.851871210 0.365189739  -5.07098368 3.957647e-07
## Sexmale     -3.577473658 0.243220900 -14.70874281 5.665010e-49
## Age         -0.035954886 0.008896908  -4.04127870 5.316053e-05
## SibSp       -0.393159871 0.141831354  -2.77202368 5.570898e-03
## Parch       -0.103400519 0.138329706  -0.74749323 4.547659e-01
## Fare         0.002793923 0.002383269   1.17230680 2.410739e-01
## EmbarkedQ    0.052614888 0.599312836   0.08779203 9.300420e-01
## EmbarkedS   -0.100084328 0.288446541  -0.34697704 7.286086e-01
```

detecta el efecto de cada predictor en la clasificación.

```r
significance <- summary_A$coefficients[, 4] < 0.05
significance
```

```
## (Intercept)     Pclass2     Pclass3     Sexmale         Age       SibSp
##        TRUE        TRUE        TRUE        TRUE        TRUE        TRUE
##       Parch        Fare   EmbarkedQ   EmbarkedS
##       FALSE       FALSE       FALSE       FALSE
```

```r
effects <- data.frame(
  Variable = names(coefficients_A),
  Coefficient = coefficients_A,
  Significant = significance
)
effects
```

```
##                Variable Coefficient Significant
## (Intercept) (Intercept)       4.007        TRUE
## Pclass2         Pclass2      -0.953        TRUE
## Pclass3         Pclass3      -1.852        TRUE
## Sexmale         Sexmale      -3.577        TRUE
## Age                 Age      -0.036        TRUE
## SibSp             SibSp      -0.393        TRUE
## Parch             Parch      -0.103       FALSE
## Fare               Fare       0.003       FALSE
## EmbarkedQ     EmbarkedQ       0.053       FALSE
## EmbarkedS     EmbarkedS      -0.100       FALSE
```

La ecuación del modelo logístico nos muestra que las variables más significativas para predecir la supervivencia son: `Pclass2`, `Pclass3`, `Sexmale`, `Age` y `SibSp`, todas con un $p < 0.05$ y las variables :`Parch`, `Fare`, `EmbarkedQ`, y `EmbarkedS` no son significativas $p > 0.05$ y tienen un impacto mínimo en la predicción, podemos hacer las observaciones clave de:

- Clase (`Pclass`): Los pasajeros de segunda y tercera clase tienen menor probabilidad de supervivencia en comparación con la primera clase.
- Género (`Sex`): Ser hombre reduce significativamente la probabilidad de supervivencia (tienes mayor posibilidad de supervivencia si eres mujer.
- Edad (`Age`): Un aumento en la edad disminuye ligeramente la probabilidad de supervivencia.
-

Familiares a bordo (`SibSp`): Tener más familiares (esposos/hermanos) a bordo reduce la probabilidad de supervivencia.

4. Analiza las predicciones para los datos de entrenamiento

Elabora la matriz de confusión

```
predicted_probs <- predict(A, newdata = M_train_clean, type = "response")
predicted_classes <- ifelse(predicted_probs > 0.5, 1, 0)

library(caret)
confusion_matrix <- confusionMatrix(
  factor(predicted_classes, levels = c(0, 1)),
  factor(M_train_clean$Survived, levels = c(0, 1))
)
confusion_matrix
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 404  65
##          1  46 224
##
##                Accuracy : 0.8498
##                  95% CI : (0.822, 0.8748)
##     No Information Rate : 0.6089
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.6809
##
##  Mcnemar's Test P-Value : 0.08755
##
##             Sensitivity : 0.8978
##             Specificity : 0.7751
##          Pos Pred Value : 0.8614
##          Neg Pred Value : 0.8296
##              Prevalence : 0.6089
##          Detection Rate : 0.5467
##    Detection Prevalence : 0.6346
##       Balanced Accuracy : 0.8364
##
```

```
##        'Positive' Class : 0
##
```

Elabora la Curva ROC

```
library(pROC)
```

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

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

```
##
## Adjuntando el paquete: 'pROC'
```

```
## The following object is masked from 'package:Metrics':
##
##     auc
```

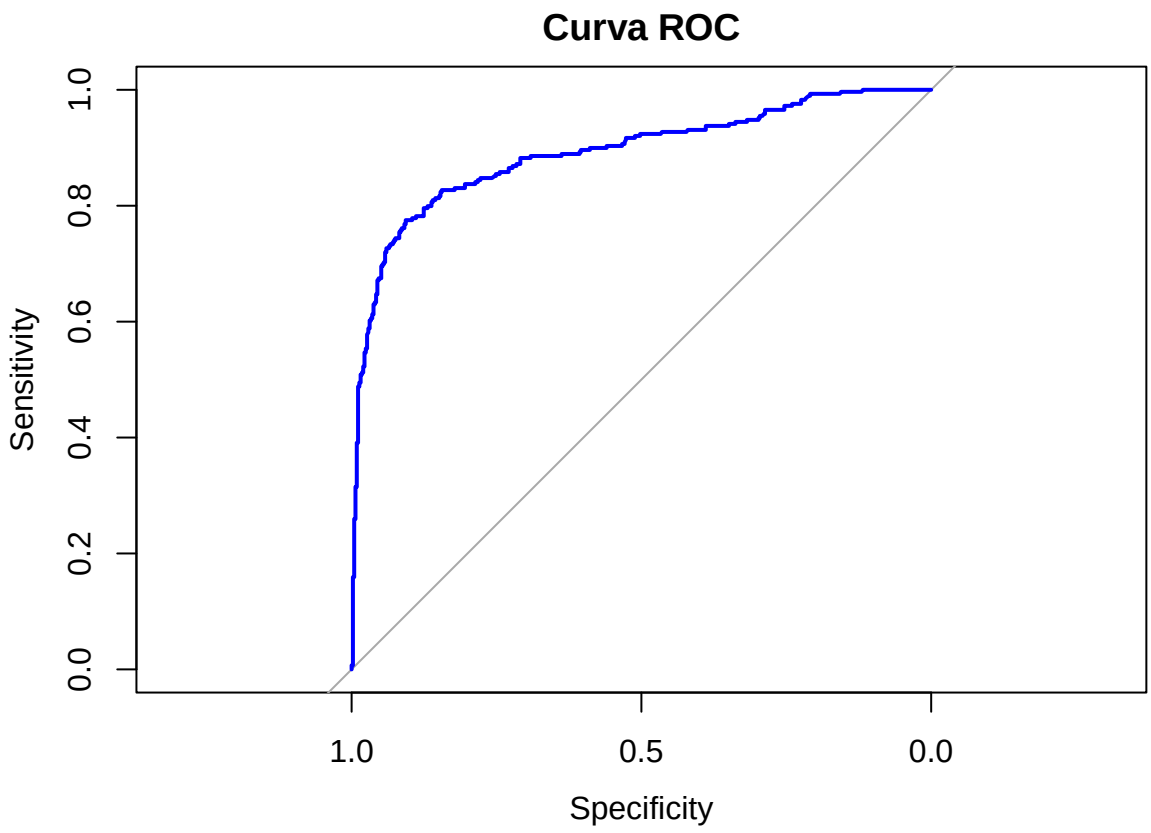
```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
roc_curve <- roc(M_train_clean$Survived, predicted_probs)
```

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

```
## Setting direction: controls < cases
```

```
plot(roc_curve, col = "blue", main = "Curva ROC")
```

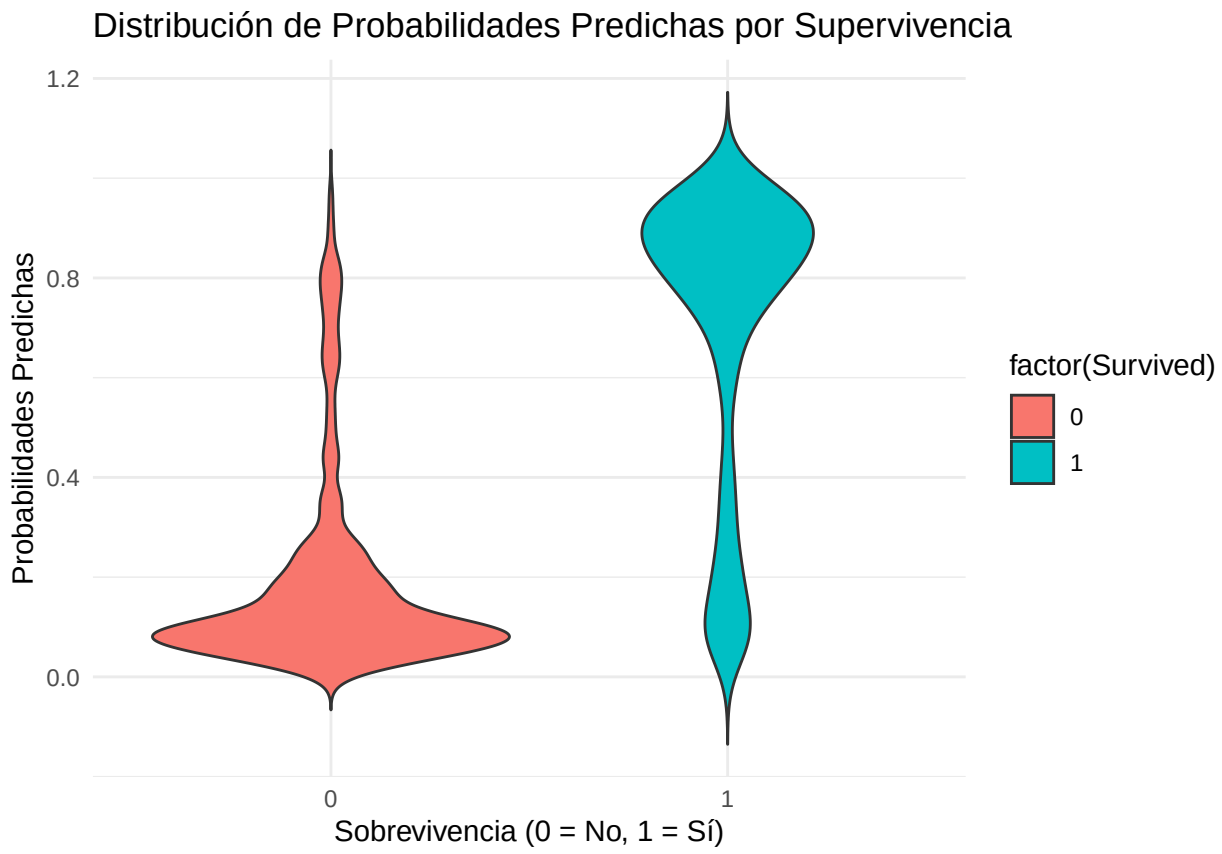



```
auc(roc_curve)
```

```
## Area under the curve: 0.8915
```

El gráfico de violín

```
library(ggplot2)

M_train_clean$predicted_probs <- predicted_probs
ggplot(M_train_clean, aes(x = factor(Survived), y = predicted_probs, fill = factor(Survived))) +
  geom_violin(trim = FALSE) +
  labs(title = "Distribución de Probabilidades Predichas por Supervivencia",
       x = "Sobrevivencia (0 = No, 1 = Sí)",
       y = "Probabilidades Predichas") +
  theme_minimal()
```



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

El modelo nos muestra un buen desempeño con una precisión del 84.98%, una sensibilidad del 89.78%, y una especificidad del 77.51%. Con la curva ROC vemos un buen desempeño discriminativo con un AUC alto. Y en el gráfico de violín vemos una clara separación de las probabilidades predichas entre sobrevivientes y no sobrevivientes, confirmando la capacidad del modelo para clasificar correctamente las dos clases. Con estos resultados vemos que el modelo si es confiable para predecir la supervivencia.

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

Elije un umbral de clasificación óptimo

```
M_valid_clean <- na.omit(M_valid)

predicted_probs_valid <- predict(A, newdata = M_valid_clean, type = "response")

library(pROC)
roc_curve_valid <- roc(M_valid_clean$Survived, predicted_probs_valid)
```

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

```
## Setting direction: controls < cases
```

```
optimal_threshold <- coords(roc_curve_valid, "best", ret = "threshold")
optimal_threshold
```

```
##   threshold
## 1 0.2418244
```

Elabora la matriz de confusión con el umbral de clasificación óptimo

```
summary(predicted_probs_valid)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00548 0.08826 0.19707 0.38385 0.77884 0.97271
```

```
threshold_adjusted <- 0.5

predicted_classes_valid <- ifelse(predicted_probs_valid > threshold_adjusted, 1, 0)

predicted_classes_valid <- factor(predicted_classes_valid, levels = c(0, 1))
actual_classes_valid <- factor(M_valid_clean$Survived, levels = c(0, 1))

library(caret)
confusion_matrix_valid <- confusionMatrix(predicted_classes_valid, actual_classes_valid)
print(confusion_matrix_valid)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 163  31
##          1  15  95
##
##                Accuracy : 0.8487
##                  95% CI : (0.8034, 0.887)
##     No Information Rate : 0.5855
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.6824
##
##  Mcnemar's Test P-Value : 0.02699
##
##             Sensitivity : 0.9157
##             Specificity : 0.7540
##          Pos Pred Value : 0.8402
##          Neg Pred Value : 0.8636
##              Prevalence : 0.5855
##          Detection Rate : 0.5362
##    Detection Prevalence : 0.6382
##       Balanced Accuracy : 0.8348
```

```
##
##        'Positive' Class : 0
##
```

El modelo tiene una precisión del 84.87%, alta sensibilidad 91.57% y buena concordancia Kappa = 0.68, aunque podría mejorar en especificidad 75.40%**, el modelo es eficaz para predecir sobrevivientes.

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

```
M_test_clean <- M_test[complete.cases(M_test), ]

M_test_clean$Pclass <- factor(M_test_clean$Pclass)
```

```
predicted_probs_test <- predict(A, newdata = M_test_clean, type = "response")
predicted_classes_test <- ifelse(predicted_probs_test > 0.9, 1, 0)
predicted_classes_test
```

```
##   1   2   3   4   5   6   7   8   9  10  12  13  14  15  16  17  18  19  20  21
##   0   0   0   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   0
##  22  24  25  26  27  28  29  31  32  33  35  36  38  39  41  43  44  45  46  47
##   0   0   1   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##  49  50  51  52  53  54  56  57  58  60  61  62  63  64  65  67  68  69  70  71
##   0   0   0   0   0   1   0   0   0   1   0   0   0   0   0   0   0   0   0   0
##  72  73  74  75  76  78  79  80  81  82  83  87  88  90  91  93  95  96  97  98
##   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   1   0   0   0   0
##  99 100 101 102 104 105 106 107 110 111 113 114 115 116 118 119 120 121 123 124
##   0   0   1   0   0   0   0   0   0   0   1   0   0   0   0   0   0   1   1   0
## 126 127 129 130 131 132 135 136 137 138 139 140 141 142 143 144 145 146 148 150
##   0   0   0   0   0   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0
## 151 154 155 156 157 158 159 160 162 163 165 166 167 168 170 172 173 175 176 177
##   1   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   1   1
## 178 179 180 181 182 183 185 186 187 188 190 191 193 194 195 196 197 198 199 202
##   0   0   0   0   0   1   1   0   0   0   0   0   0   0   0   0   0   0   0   0
## 203 204 205 207 208 209 210 211 213 214 215 216 218 219 221 222 223 224 225 227
##   0   1   0   0   0   1   0   0   0   0   0   0   0   1   0   0   0   0   0   0
## 229 230 231 232 233 235 236 237 238 239 240 241 242 243 246 247 248 249 251 252
##   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   1   0   0   1   0
## 253 254 255 258 259 260 261 262 263 264 265 270 271 273 276 277 278 279 280 281
##   0   0   0   0   1   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0
## 282 284 285 286 288 292 294 295 296 297 299 300 301 303 304 306 307 308 309 310
##   0   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0
## 311 312 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331
##   0   0   0   1   0   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0
## 332 334 335 336 337 338 339 341 342 344 346 347 348 349 350 351 352 353 354 355
##   0   0   0   0   0   0   0   0   0   1   0   0   0   0   0   1   0   0   0   0
## 356 357 360 361 362 363 364 365 368 369 370 371 372 373 374 375 376 377 378 379
##   0   0   0   0   0   0   0   1   0   1   0   0   1   0   0   0   1   0   0   0
## 380 382 384 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402
##   0   0   0   0   0   0   0   0   0   0   0   0   0   1   0   0   0   0   1   0
## 403 404 405 406 407 408 410 412 413 415 416
##   1   0   0   0   0   0   0   1   0   1   0
```

```r
M_test_clean$Survived <- M$Survived[1:nrow(M_test_clean)]
```

```r
actual_classes_test <- M_test_clean$Survived
correct_predictions <- sum(predicted_classes_test == actual_classes_test)
accuracy <- (correct_predictions / length(actual_classes_test)) * 100
print(paste("Precisión de las predicciones:", accuracy, "%"))
```

```
## [1] "Precisión de las predicciones: 60.4229607250755 %"
```

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?

Las principales características que influyen en la supervivencia en el Titanic incluyen variables como:

-Pclass: La clase de boletos (1, 2, o 3) influye directamente en las probabilidades de sobrevivir ya que las personas que se encontraban en primera clase tendrían más probabilidades de sobrevivir que las de la tercera.

Sex: El género es una variable muy relevante en este caso poruqe las mujeres tuvieron más probabilidades de sobrevivir debido a las políticas de evacuación de 'niños y mujeres primero'.

Age: La edad también fue un gran factori que influyó en la supervivencia dado que los niños y personas jóvenes tienen más probabilidades de sobrevivir en este caso.

Fare: El precio del boleto (Fare) es indicativo del acceso a mejores clases, que a su vez están correlacionadas con mayores probabilidades de sobrevivir.

Embarked: El puerto de embarque también puede ser relevante, pues algunos puertos podrían estar más directamente relacionados con las clases de pasajeros que como se explicó en Pclass influye directamente en las probabilidades de sobrevivir.

Interpreta los coeficientes del modelo

```r
summary(A)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp + Parch +
##     Fare + Embarked, family = "binomial", data = M_train_clean)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.007401   0.548873   7.301 2.85e-13 ***
## Pclass2     -0.953392   0.352453  -2.705  0.00683 **
## Pclass3     -1.851871   0.365190  -5.071 3.96e-07 ***
## Sexmale     -3.577474   0.243221 -14.709  < 2e-16 ***
## Age         -0.035955   0.008897  -4.041 5.32e-05 ***
## SibSp       -0.393160   0.141831  -2.772  0.00557 **
## Parch       -0.103401   0.138330  -0.747  0.45477
## Fare         0.002794   0.002383   1.172  0.24107
## EmbarkedQ    0.052615   0.599313   0.088  0.93004
```

```
## EmbarkedS   -0.100084   0.288447  -0.347  0.72861
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 989.11  on 738  degrees of freedom
## Residual deviance: 564.63  on 729  degrees of freedom
## AIC: 584.63
##
## Number of Fisher Scoring iterations: 5
```

Los coeficientes más rlevantes para la supervivencia son los siguientes:

-Sexmale: Los hombres tienen una probabilidad significativamente más baja de sobrevivir, con un coeficiente de -3.57.

-Pclass: Los pasajeros que pertenecian a clases más bajas (Pclass2 y Pclass3) tienen muchas menos probabilidad de sobrevivir, los coeficientes negativos para Pclass2 (-0.95) y Pclass3 (-1.85) indican que la clase baja está fuertemente asociada con una menor supervivencia.

-Age: Conforme más edad se tenga la probabilidad de supervivencia disminuye ligeramente (coeficiente de -0.036).

-Fare: Pagar un precio más alto está asociado con una mayor probabilidad de supervivencia (coeficiente de 0.0028)

Define cuál es el mejor umbral de clasificación y por qué

El umbral óptimo va a depender de cómo se quiere balancear la precisión y la recuperación. Usualmente se usa un umbral de 0.5 como punto de partida, sin embargo si quiesieramos maximizar la precisión o la recuperación lo podemos ir ajustando de acuerdo a las métricas de desempeño que calcules, como la matriz de confusión, la curva ROC, como se hizo en este caso.