Regresión Logística. El titanic

Nombre del estudiante

2024-11-19

Bibliotecas

```
# Cargamos todas las librería en la lista "librerias"
librerias = c('tidyverse','broom','ISLR','GGally','modelr','cowplot','rlang','modelr','tibble','Metrics

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

```
## -- 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 error
## 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
##
##
##
## Attaching package: 'Metrics'
##
##
## The following object is masked from 'package:rlang':
##
##     ll
##
##
## The following objects are masked from 'package:modelr':
##
##     mae, mape, mse, rmse
##
##
##
## Attaching package: 'mice'
##
##
## The following object is masked from 'package:stats':
##
##     filter
##
##
## The following objects are masked from 'package:base':
##
##     cbind, rbind
##
##
## 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
```

Leyendo los datos:

```
M = read.csv("Titanic.csv")
str(M)
```

```
## 'data.frame':    1309 obs. of  12 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Survived   : int  0 1 0 0 1 0 1 0 1 0 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : chr  "Kelly, Mr. James" "Wilkes, Mrs. James (Ellen Needs)" "Myles, Mr. Thomas Francis
##  $ Sex        : chr  "male" "female" "male" "male" ...
##  $ 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 ...
##  $ Ticket     : chr  "330911" "363272" "240276" "315154" ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : chr  "" "" "" "" ...
##  $ Embarked   : chr  "Q" "S" "Q" "S" ...
```

Las variables son:

- *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)

Preparación de la base de datos

Ajustando las variables

Variables de interés: Quita aquellas que de entrada no tengan que ver con la sobrevivencia del pasajero. Por ejemplo: Quitar variables 4, 9 y 11 (define si hay más)

Variables categóricas que deben aparecer como factores: define qué variables aparecerán como factores Por ejemplo: Survived, Pclass, Sex y Embarked (define si hay más)

```
# Eliminar variables:
M1 <- M[,c(-4,-9,-11)]

#Transformar a factores:
for(var in c('Survived','Pclass','Embarked','Sex'))
  M1[,var] <-as.factor(M1[,var])
```

Análisis de 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(M1,M1[,i])=="")}
V
```

0
0
0
0
NA
0
0
NA
NA

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

Para contar los datos faltantes:

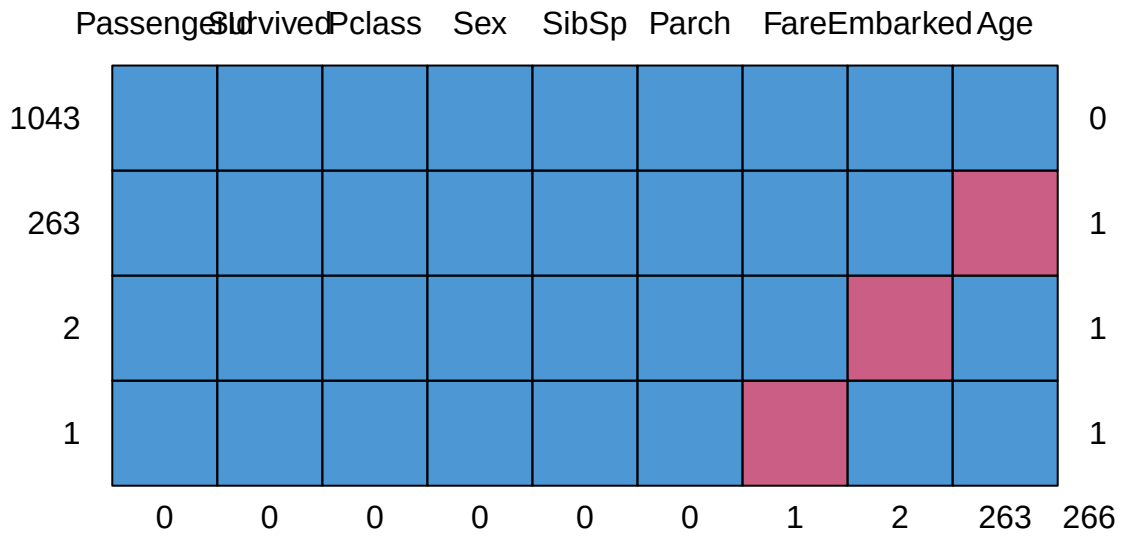
```
N = apply(X=is.na(M1),MARGIN = 2,FUN = sum)
P = round(100*N/length(M1[,2]),2)
NP = data.frame(as.numeric(N),as.numeric(P))
row.names(NP)= c("PassengerId", "Survived", "Pclass", "Sex", "Age", "SibSp", "Parch", "Fare", "Embarked
names(NP)=c("Número","Porcentaje")
t(NP)
```

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
Número	0	0	0	0	263.00	0	0	1.00	2.00
Porcentaje	0	0	0	0	20.09	0	0	0.08	0.15

En edad hay muchos datos faltantes, el 20% de los datos.

Observemos el patrón de los datos faltantes:

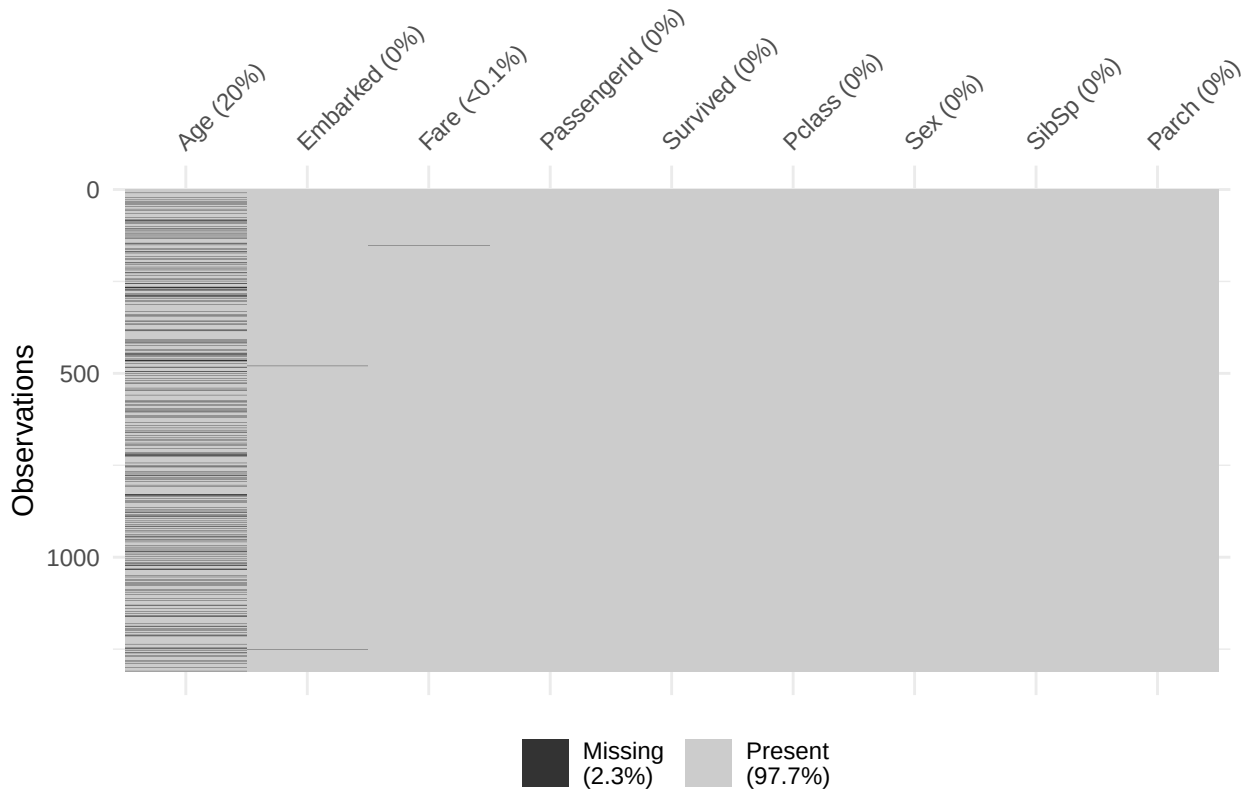
```
md.pattern(M1)
```



	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

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(M1,sort_miss = TRUE)
```



Análisis sobre datos faltantes

Medidas con datos faltantes

```
summary(M1[,-1])
```

Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0:815	1:323	female:466	Min. : 0.17	Min. :0.0000	Min. :0.000	Min. : 0.000	C :270
1:494	2:277	male :843	1st Qu.:21.00	1st Qu.:0.0000	1st Qu.:0.000	1st Qu.: 7.896	Q :123
NA	3:709	NA	Median :28.00	Median :0.0000	Median :0.000	Median : 14.454	S :914
NA	NA	NA	Mean :29.88	Mean :0.4989	Mean :0.385	Mean : 33.295	NA's: 2
NA	NA	NA	3rd Qu.:39.00	3rd Qu.:1.0000	3rd Qu.:0.000	3rd Qu.: 31.275	NA
NA	NA	NA	Max. :80.00	Max. :8.0000	Max. :9.000	Max. :512.329	NA
NA	NA	NA	NA's :263	NA	NA	NA's :1	NA

Medidas sin datos faltantes

```
M2 = na.omit(M1)
summary(M2[,-1])
```

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

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

Si difieren, la cantidad de las personas cambia en gran cantidad y los valores de "Embarked" que se reducen de gran mmanera, esto debido a que en esa categoria habia un gran numero de datos faltantes dentro de la misma, mientras que aumenta "Fare" debido a la eliminacion de los datos vacios.

Sobrevivientes

```
t2c = 100*prop.table(table(M1[,2]))
t2s = 100*prop.table(table(M2[,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(M1[,3]))
t3s = 100*prop.table(table(M2[,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

	Con NA (%)	Sin NA (%)	Pérdida (prop)
Tercera	54.16	47.94	0.89

Sexo

```
t4c = 100*prop.table(table(M1[,4]))
t4s = 100*prop.table(table(M2[,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(M1[,9]))
t9s = 100*prop.table(table(M2[,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

En este ensayo quitarás los datos faltantes, pero deberás indicar cuáles son las variables más afectadas y por qué.

Análisis descriptivo

Se recomienda analizar dividiendo la base de datos entre los que sobrevivieron y los que no. Usa:

- Medidas
- Gráficos

Partición. Entrenamiento y prueba

Se toma el 70% de la muestra como entrenamiento y el 30% para prueba.

```r
M_indice <- createDataPartition(M2$Survived, p = .7, list = FALSE, times = 1)

M_train <- M2[ M_indice,]  %>% as_tibble()
M_valid <- M2[-M_indice,] %>% as_tibble()
```

Modelación (entrenamiento)

Comienza con el modelo completo, incluyendo las variables categóricas (factores). Aplica el comando *step* para poder encontrar el mejor modelo.

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.

```r
A = glm(Survived ~., data = M_train, family = "binomial")
```

```r
step(A, direction="both", trace=1 )
```

```
## Start:  AIC=582.33
## Survived ~ PassengerId + Pclass + Sex + Age + SibSp + Parch +
##     Fare + Embarked
##
##               Df Deviance    AIC
## - Embarked     2   561.38 579.38
## - PassengerId  1   560.66 580.66
## - Fare         1   561.32 581.32
## <none>             560.33 582.33
## - Parch        1   563.16 583.16
## - SibSp        1   567.07 587.07
## - Age          1   572.38 592.38
## - Pclass       2   585.11 603.11
## - Sex          1   887.46 907.46
##
## Step:  AIC=579.38
## Survived ~ PassengerId + Pclass + Sex + Age + SibSp + Parch +
##     Fare
##
##               Df Deviance    AIC
## - PassengerId  1   561.67 577.67
## - Fare         1   562.79 578.79
## <none>             561.38 579.38
## - Parch        1   564.33 580.33
## + Embarked     2   560.33 582.33
## - SibSp        1   568.78 584.78
## - Age          1   574.07 590.07
## - Pclass       2   588.98 602.98
## - Sex          1   892.79 908.79
##
## Step:  AIC=577.67
## Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare
##
##               Df Deviance    AIC
```

```
## - Fare         1   563.09 577.09
## <none>             561.67 577.67
## - Parch        1   564.58 578.58
## + PassengerId  1   561.38 579.38
## + Embarked     2   560.66 580.66
## - SibSp        1   568.97 582.97
## - Age          1   574.32 588.32
## - Pclass       2   589.02 601.02
## - Sex          1   893.89 907.89
##
## Step:  AIC=577.09
## Survived ~ Pclass + Sex + Age + SibSp + Parch
##
##               Df Deviance    AIC
## <none>             563.09 577.09
## - Parch        1   565.27 577.27
## + Fare         1   561.67 577.67
## + PassengerId  1   562.79 578.79
## + Embarked     2   561.66 579.66
## - SibSp        1   569.73 581.73
## - Age          1   576.75 588.75
## - Pclass       2   615.55 625.55
## - Sex          1   895.74 907.74

##
## Call:  glm(formula = Survived ~ Pclass + Sex + Age + SibSp + Parch,
##     family = "binomial", data = M_train)
##
## Coefficients:
## (Intercept)      Pclass2      Pclass3      Sexmale          Age        SibSp
##     4.23820     -1.27833     -2.03081     -3.71019     -0.03126     -0.34792
##       Parch
##    -0.18612
##
## Degrees of Freedom: 730 Total (i.e. Null);  724 Residual
## Null Deviance:       982.8
## Residual Deviance: 563.1     AIC: 577.1
```

- Identifica el mejor modelo de acuerdo con el AIC
- Selecciona la última variable que elminó el comando *step*. Prueba dos modelos, uno con esa variable y otro sin ella.

El mejor modelo fue Survived ~ Pclass + Sex + Age + SibSp, y se comprobabra con el otro modelo; Survived ~ PassengerId + Pclass + Sex + Age + SibSp

Modelo B

- Prueba el modelo incluyendo la última variable que eliminó el comando *step*.
- Indica cuáles son las variables que incluye.
- Interpreta la significancia global (de todo el modelo) y la individual (de cada una de las variables)

```r
B = glm(formula = Survived ~ PassengerId + Pclass + Sex + Age + SibSp, family = "binomial", data = M_tra
summary(B)
```

```
##
## Call:
## glm(formula = Survived ~ PassengerId + Pclass + Sex + Age + SibSp,
##     family = "binomial", data = M_train)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.2175785  0.5121730   8.235  < 2e-16 ***
## PassengerId -0.0001473  0.0002886  -0.510 0.609878
## Pclass2     -1.2798903  0.3055964  -4.188 2.81e-05 ***
## Pclass3     -2.0520637  0.2947573  -6.962 3.36e-12 ***
## Sexmale     -3.6132842  0.2405031 -15.024  < 2e-16 ***
## Age         -0.0310480  0.0086206  -3.602 0.000316 ***
## SibSp       -0.4031001  0.1339922  -3.008 0.002626 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 982.8  on 730  degrees of freedom
## Residual deviance: 565.0  on 724  degrees of freedom
## AIC: 579
##
## Number of Fisher Scoring iterations: 5
```

Tiene una significancia global muy alta se puede ver en elnull deviance (982) y residual deviance (544), mientras que la individual depende, ya que se puede observar que Passenger ID no es significatiiva, mientras que Pclass2, Pclass3, Sex, Age y SibSP tiienen un impacto en la posibbildad de supervivencia de la persona.

Modelo C

- Prueba el modelo tal como te lo recomendó el comando *step*.
- Indica cuáles son las variables que incluye.
- Interpreta la significancia global (de todo el modelo) y la individual (de cada una de las variables)

```r
C = glm(formula = Survived ~ Pclass + Sex + Age + SibSp, family = "binomial", data = M_train)
summary(C)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp, family = "binomial",
##     data = M_train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.115024   0.469056   8.773  < 2e-16 ***
## Pclass2     -1.281753   0.305426  -4.197 2.71e-05 ***
## Pclass3     -2.042856   0.294188  -6.944 3.81e-12 ***
```

```
## Sexmale     -3.616349   0.240513 -15.036  < 2e-16 ***
## Age         -0.030952   0.008616  -3.592 0.000328 ***
## SibSp       -0.399850   0.133714  -2.990 0.002787 **
## ---
## 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: 565.27  on 725  degrees of freedom
## AIC: 577.27
##
## Number of Fisher Scoring iterations: 5
```

El modelo tiene una significancia global muy alta, como se puede observar en la null deviance (982.80) y la residual deviance (544.74), a su vez, la significancia individual, se observa que las variables Pclass2, Pclass3, Sex, Age y SibSp son todas significativas. Esto indica que estas variables tienen un impacto claro en la probabilidad de supervivencia.

Análisis de los modelos B y C

Resumen de los indicadores importantes de los modelos B y C

Compara el AIC, la *Null Deviance* y la *Residual Deviance* de los modelos B y C. Extrae los valores con los modelos con los comandos:

- B$aic
- B$deviance
- B$null.deviance

```r
B_aic <- B$aic
B_deviance <- B$deviance
B_null_deviance <- B$null.deviance

C_aic <- C$aic
C_deviance <- C$deviance
C_null_deviance <- C$null.deviance

tabla_comparativa <- data.frame(
  Indicador = c("AIC", "Residual Deviance", "Null Deviance"),
  Modelo_B = c(B_aic, B_deviance, B_null_deviance),
  Modelo_C = c(C_aic, C_deviance, C_null_deviance)
)

print(tabla_comparativa)
```

```
##           Indicador Modelo_B Modelo_C
## 1               AIC 579.0048 577.2653
## 2 Residual Deviance 565.0048 565.2653
## 3     Null Deviance 982.7966 982.7966
```

Elabora una tabla comparativa ¿Cómo se comporta la *Null Deviance*? ¿por qué? ¿Qué pasa con el AIC y la *Residual Deviance*?

No cambia la Null deviance debido a que amos modelos tienen las mismas variables,tambien, se puede ver que el modelo C tiene un Menor AIC, por lo que se considera que se ajusta mejor a los datos y por ultimo, se pued observar que el Resiidual Deviance son casi iguales.

Cálculo de la Desviación explicada ($pseudor^2$)

Calcula la desviación explicada para cada modelo. Recuerda que es igual a: pseudo r^2 = 1-Desviación residual/Desviación nula Compara los resultados obtenidos por ambos modelos

```
# Calcular la Desviación Explicada para el Modelo B
null_deviance_B <- B$null.deviance
residual_deviance_B <- B$deviance
desviacion_explicada_B <- (1 - (residual_deviance_B / null_deviance_B)) * 100

# Calcular la Desviación Explicada para el Modelo C
null_deviance_C <- C$null.deviance
residual_deviance_C <- C$deviance
desviacion_explicada_C <- (1 - (residual_deviance_C / null_deviance_C)) * 100

# Imprimir los resultados
cat("Desviación Explicada del Modelo B: ", round(desviacion_explicada_B, 2), "%\n")
```

```
## Desviación Explicada del Modelo B:  42.51 %
```

```
cat("Desviación Explicada del Modelo C: ", round(desviacion_explicada_C, 2), "%\n")
```

```
## Desviación Explicada del Modelo C:  42.48 %
```

Prueba de razón de verosimilitud

H_0 : El modelo con predictores explica mejor la variable respuesta: $log(\frac{p}{1-p})$ que el modelo nulo
H_1 : El modelo nulo explica mejor la variable respuesta: $log(\frac{p}{1-p})$ (la probabilidad es constante)

Se calcula el estadístico de χ^2 para la razón de verosimilutud a partir de las *Deviance* de los modelos.

```
Diferencia = C$null.deviance-B$deviance
gl = C$df.null - C$df.deviance

pchisq(Diferencia,gl,lower.tail = FALSE)
```

```
## numeric(0)
```

Interpreta en el contexto del problema No se otuvo valor P, lo cual nos indica que no se pudo realizar la comparacion entre los modelos.

Comparación entre los modelos B y C

Se pueden comparar los modelo B y C para ver si hay una diferencia significativa entre ambos con la misma razón de verosimilitud utilizando el comando ANOVA y la prueba LR.

```r
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
```

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

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
724	565.0048	NA	NA	NA
725	565.2653	-1	-0.2605018	0.6097756

Modelo Seleccionado

Define los coeficientes del modelo seleccionado. Por ejemplo, si el modelo seleccionado fue el B:

```r
b0 = round(B$coefficients[1],3)
b1 = round(B$coefficients[2],3)
b2 = round(B$coefficients[3],3)
b3 = round(B$coefficients[4],3)
b4 = round(B$coefficients[5],3)
b5 = round(B$coefficients[6],3)
b6 = round(B$coefficients[7],3)
b0
```

```
## (Intercept)
##       4.218
```

```r
b1
```

```
## PassengerId
##           0
```

```r
b2
```

```
## Pclass2
##   -1.28
```

```
b3
```

```
## Pclass3
##  -2.052
```

```
b4
```

```
## Sexmale
##  -3.613
```

```
b5
```

```
##    Age
## -0.031
```

```
b6
```

```
##  SibSp
## -0.403
```

```
b7 = round(C$coefficients[1],3)
b8 = round(C$coefficients[2],3)
b9 = round(C$coefficients[3],3)
b10 = round(C$coefficients[4],3)
b11 = round(C$coefficients[5],3)
b12 = round(C$coefficients[6],3)
b7
```

```
## (Intercept)
##       4.115
```

```
b8
```

```
## Pclass2
##  -1.282
```

```
b9
```

```
## Pclass3
##  -2.043
```

```
b10
```

```
## Sexmale
##  -3.616
```

```
b11
```

```
##    Age
## -0.031
```

```
b12
```

```
## SibSp
##  -0.4
```

Gráfica el modelo

Para percibir el efecto de cada variable, grafica cada variable contra los valores predichos por el modelo. Aunque en el modelo, la variable respuesta es:

$$\hat{y} = log\left(\frac{p}{1-p}\right)$$

con el subcomando: *fitted.values* del comando *glm* se obtienen las probabilidades estimadas para los valores datos. R despeja las probabilidades:

$$\hat{p} = \left(\frac{e^{\hat{y}}}{1+e^{\hat{y}}}\right)$$

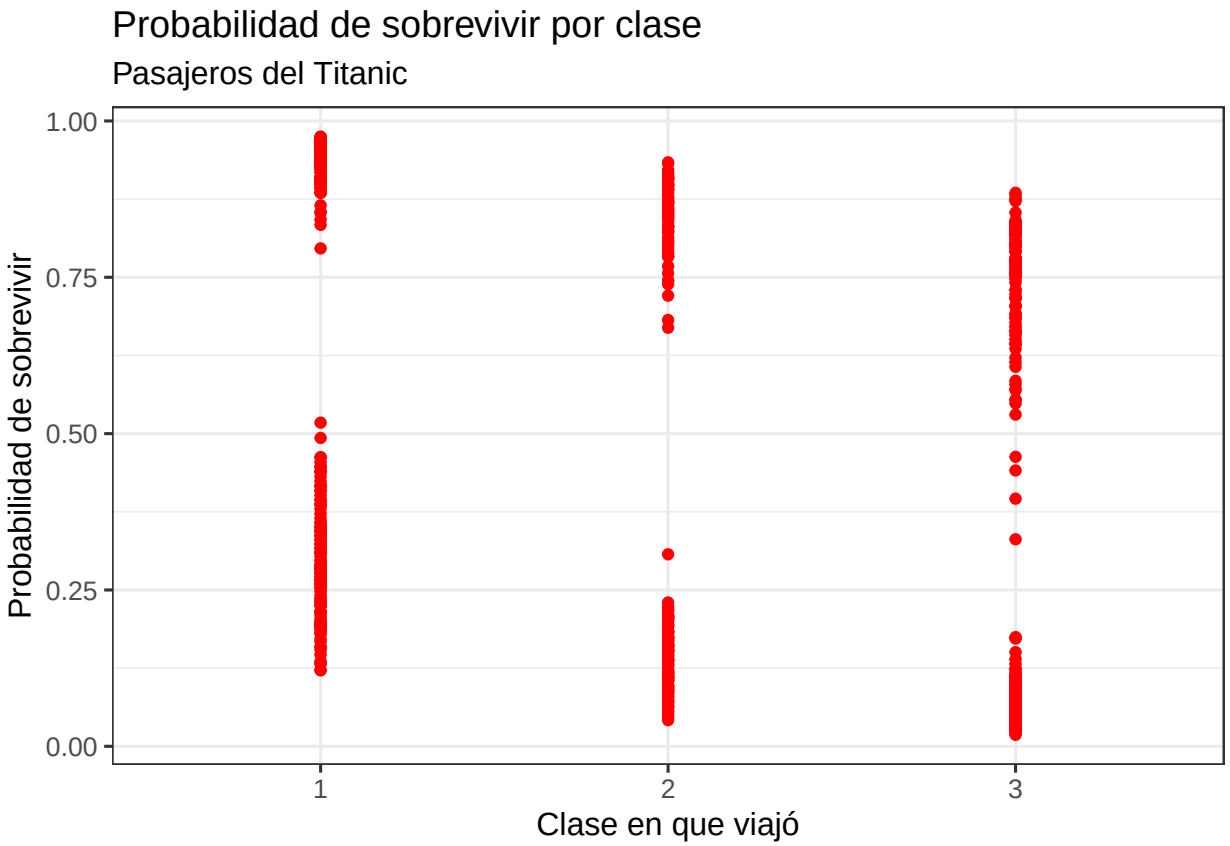
Así que interpretar el efecto de cada variable, se grafica cada una de ellas contra los valores predichos para la probabilidad de sobrevivencia.

Para hacer los gráficos se ejemplifica con:

```
p_pred = C$fitted.values
M_pred = data.frame(M_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)
```

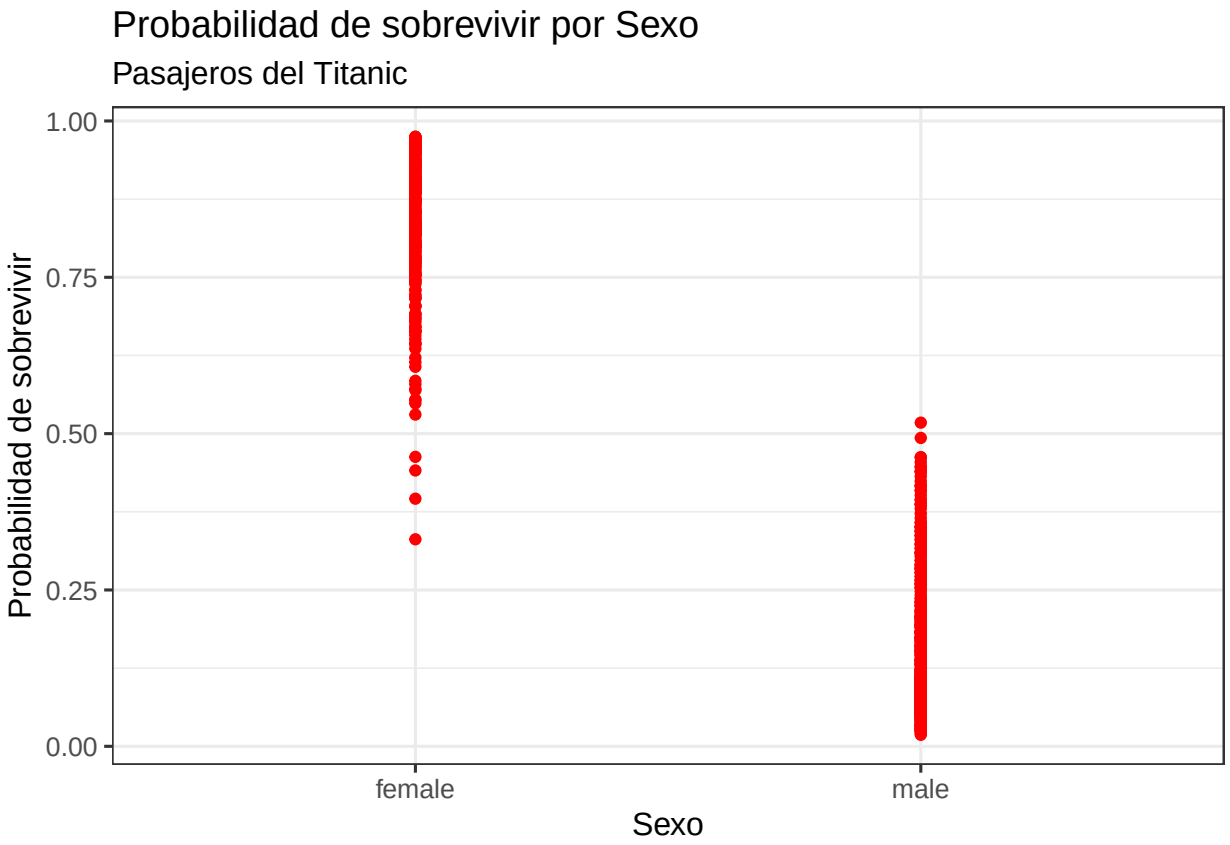
Clase en que viajó el pasajero

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



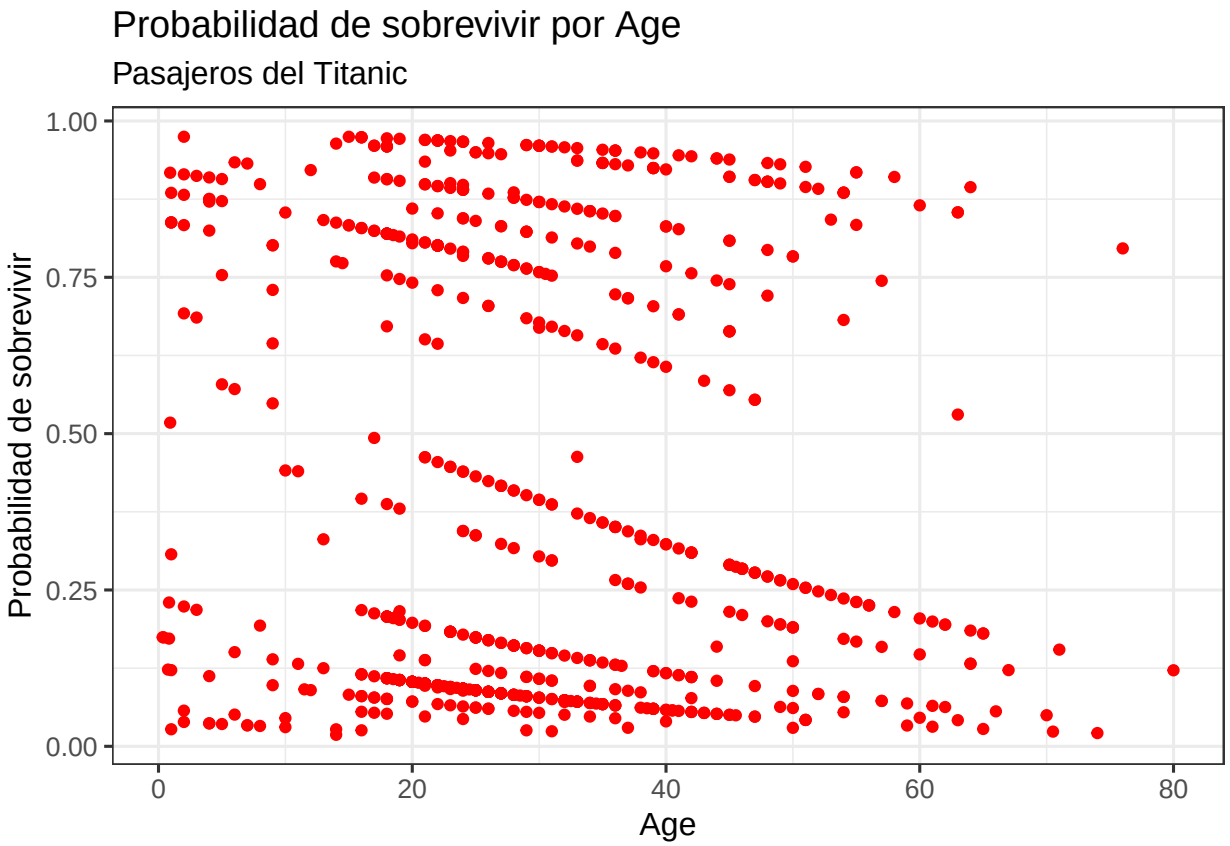
En esta grafica se puede observar que los de primera clase tenian la mayor probabilidad de sorevivir, segunda y tercera clase seguian detras, tercera siendo la clase donde se ampliaba mas la probabilidad de supervivencia.

```
p_pred2 = C$fitted.values
M_pred2 = data.frame(M_train[,c(2,3,4,5,6)],p_pred2)

ggplot(M_pred, aes( x = Sex)) +
geom_point(aes(y=M_pred2$p_pred2), size=1.5,color="red") +
  labs(x="Sexo", y="Probabilidad de sobrevivir",
       title="Probabilidad de sobrevivir por Sexo",
       subtitle="Pasajeros del Titanic",
       col="")+
  theme_bw(base_size = 12)
```



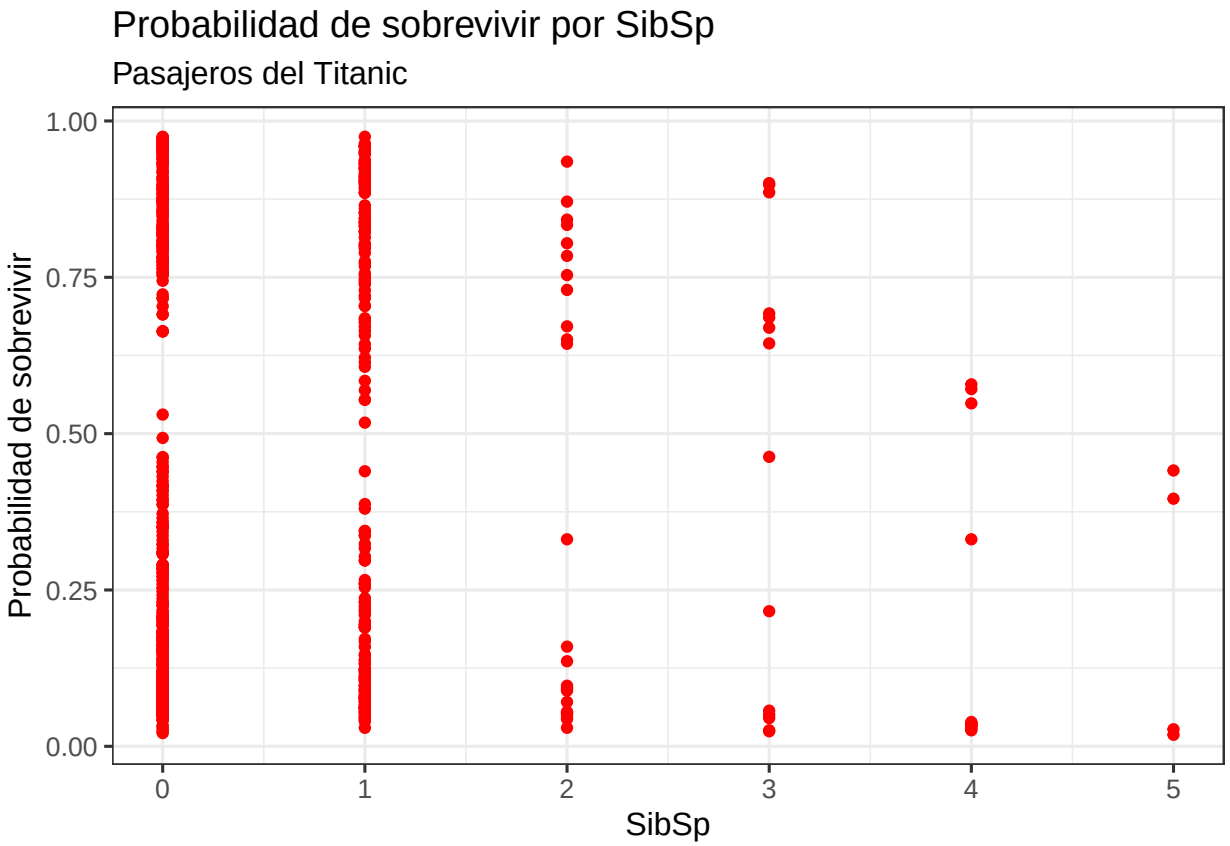
La grafica anterior nos muestra como el ser mujer aumentaba en gran medida las probabilidades de supervivencia, mientras que el ser hombre demuestra lo reducido de las mismas.

```
p_pred3 = C$fitted.values
M_pred3 = data.frame(M_train[,c(2,3,4,5,6)],p_pred3)

ggplot(M_pred, aes( x = Age)) +
geom_point(aes(y=M_pred3$p_pred3), size=1.5,color="red") +
  labs(x="Age", y="Probabilidad de sobrevivir",
       title="Probabilidad de sobrevivir por Age",
       subtitle="Pasajeros del Titanic",
       col="")+
  theme_bw(base_size = 12)
```



Se puede observar que las personas de un rangoo aproximado de 20 a 40 años eran los mas propensos a sobrevivir, a comparacion de los adultos mayores o niños.

```
p_pred4 = C$fitted.values
M_pred4 = data.frame(M_train[,c(2,3,4,5,6)],p_pred4)

ggplot(M_pred, aes( x = SibSp)) +
geom_point(aes(y=M_pred4$p_pred4), size=1.5,color="red") +
  labs(x="SibSp", y="Probabilidad de sobrevivir",
       title="Probabilidad de sobrevivir por SibSp",
       subtitle="Pasajeros del Titanic",
       col="")+
  theme_bw(base_size = 12)
```



La cantidad de SibSp tambien muestra un gran impacto en la probabilidad de supervivencia, mientras menor numeroo de familia se tenga, mas probable es que uno sobreviva.

Predicciones

Se hace el análisis con el modelo seleccionado, en el ejemplo suponemos que se seleccionó el modelo B.

Matriz de confusión

```
library(vcd)
```

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

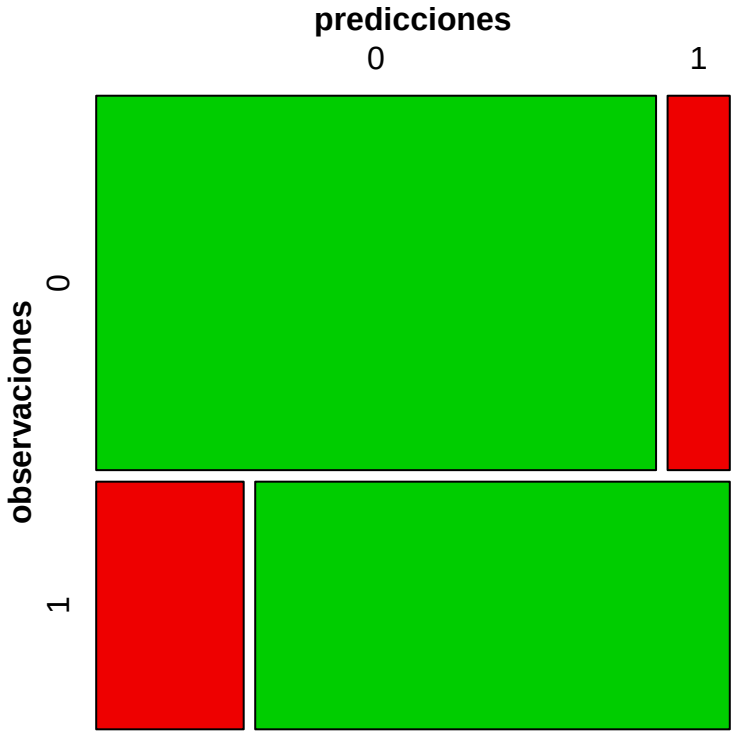
```
##
## Attaching package: 'vcd'
```

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

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

observaciones/predicciones	0	1
0	396	44
1	69	222

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




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

```
## La Exactitud (accuracy) del modelo es 0.8454172
```

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

```
## La Sensibilidad del modelo es 0.9
```

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

```
## La Especificidad del modelo es 0.7628866
```

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

```
## La Precisión del modelo es 0.8516129
```

Define si el modelo es bueno o no. El modelo es bueno ya que los resultados anteriores nos ayudan a confirmar que es equilibrado, al mismo tiemmpo que nos muestran que tiiene un buen rendimiento a la hora de predecir.

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.

```r
pred = predict(C, data = M_train, type = 'response')

library(pROC)
```

```
## 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
```

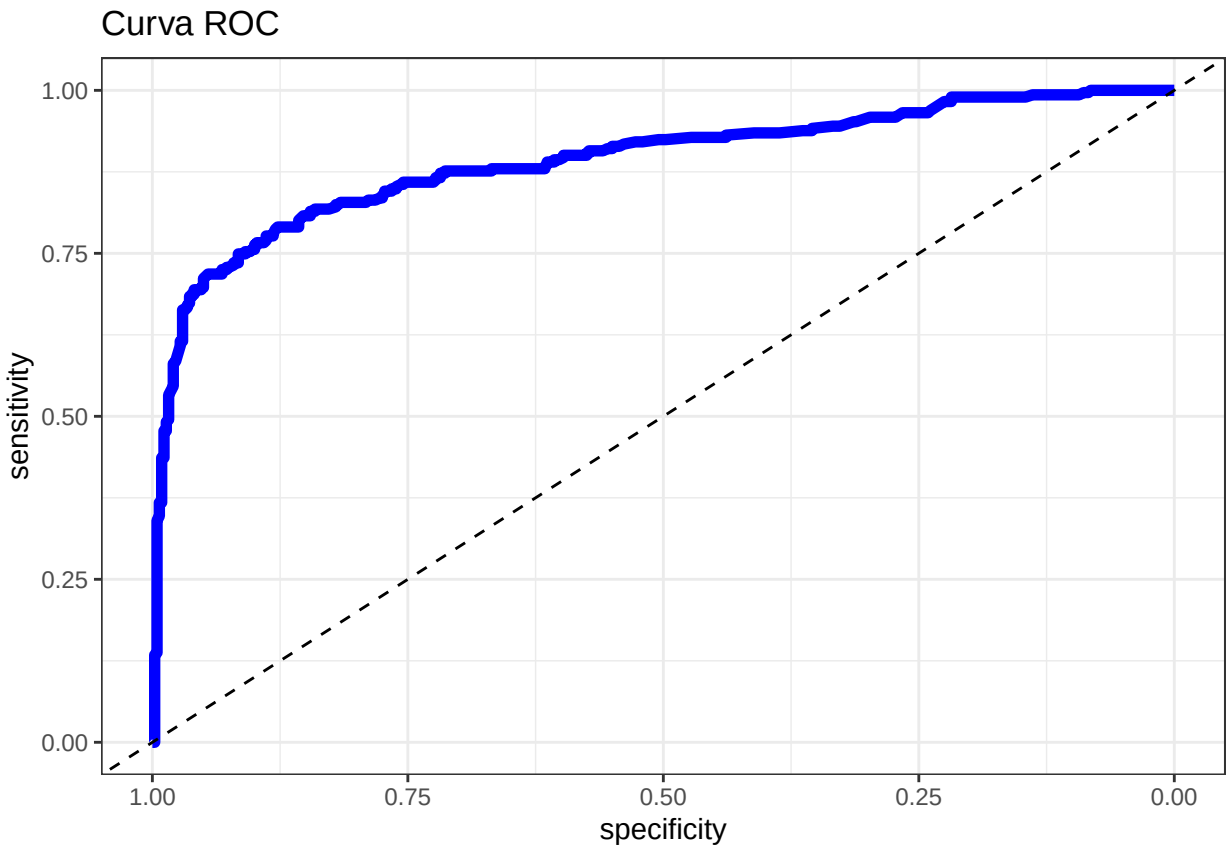
```r
ROC <- roc(response=M_train$Survived, predictor=pred)
```

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

## Setting direction: controls < cases
```

```r
ROC
```

```
##
## Call:
## roc.default(response = M_train$Survived, predictor = pred)
##
## Data: pred in 440 controls (M_train$Survived 0) < 291 cases (M_train$Survived 1).
## Area under the curve: 0.8917
```

```r
ggroc(ROC, color = "blue", size = 2) + geom_abline(slope = 1, intercept = 1, linetype ='dashed') + labs
```



Nota: Se grafica Especificidad, pero en realidad se está graficando 1 - Especificidad.

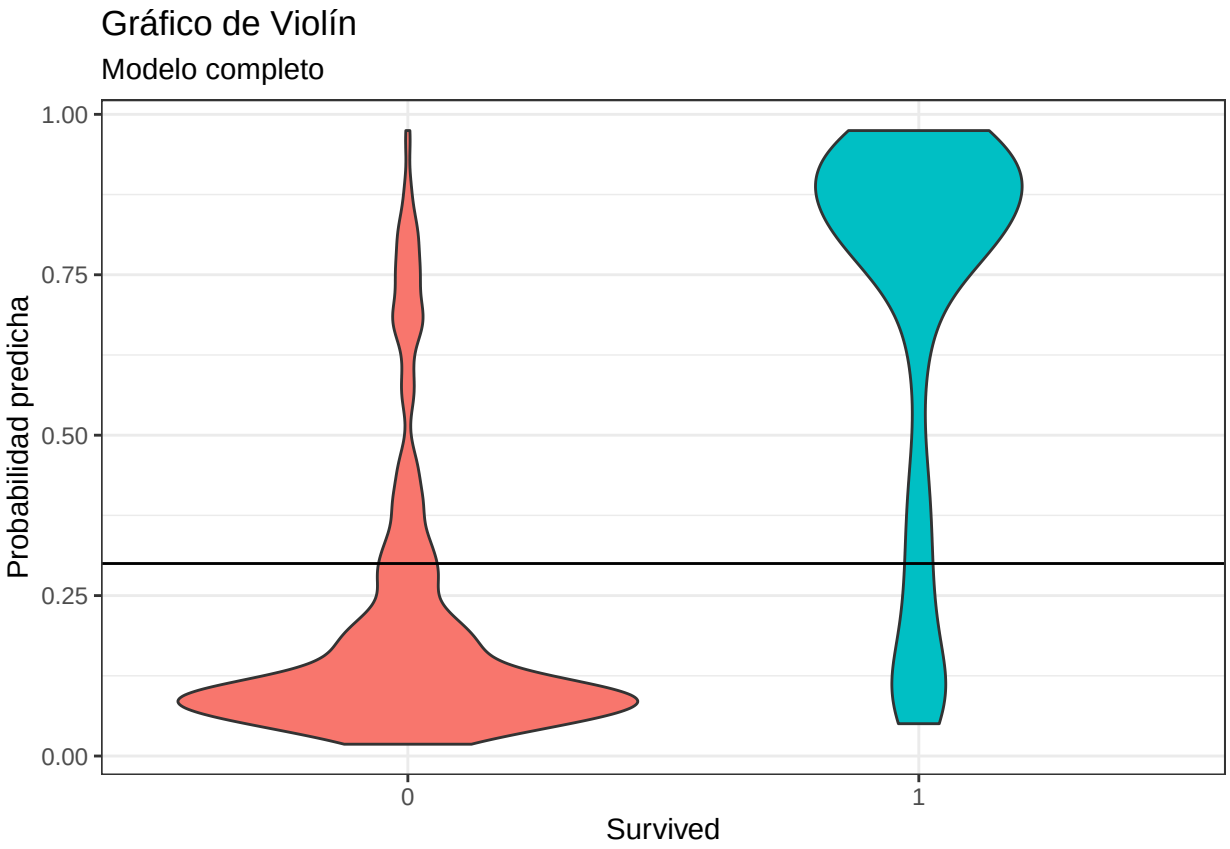
Interpreta el gráfico y la salida que da el comando *roc* Que la curva ROC este cerca de la area superior izquierda nos demuestra que esta realizando correctamente la clasificacion de los datos.

Gráfico de violín

Se crea la base de datos para el gráfico, se usan las predicciones ya elaboradas para el gráfico ROC y las clasificaciones originales (*train$M_Survived*).

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

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



Interpreta

Podemos ver que se distinguen entre las dos, donde es posible ver claramanete que hay una diferencia entree las personas que sobrevivieron y las que no.

Validación

Elección de un umbral de clasificación optimo.

Elección del umbral de clasificación (punto de corte)

Se trabaja con la base de datos de validación (*M_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 (en el ejemplo, el B)
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(C, newdata=M_valid, type='response')
clase_real = M_valid$Survived
```

```r
datosV = data.frame(accuracy=NA, recall=NA, specificity = NA, precision=NA)

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 positivos
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'.

```r
library(reshape2)
```

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

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

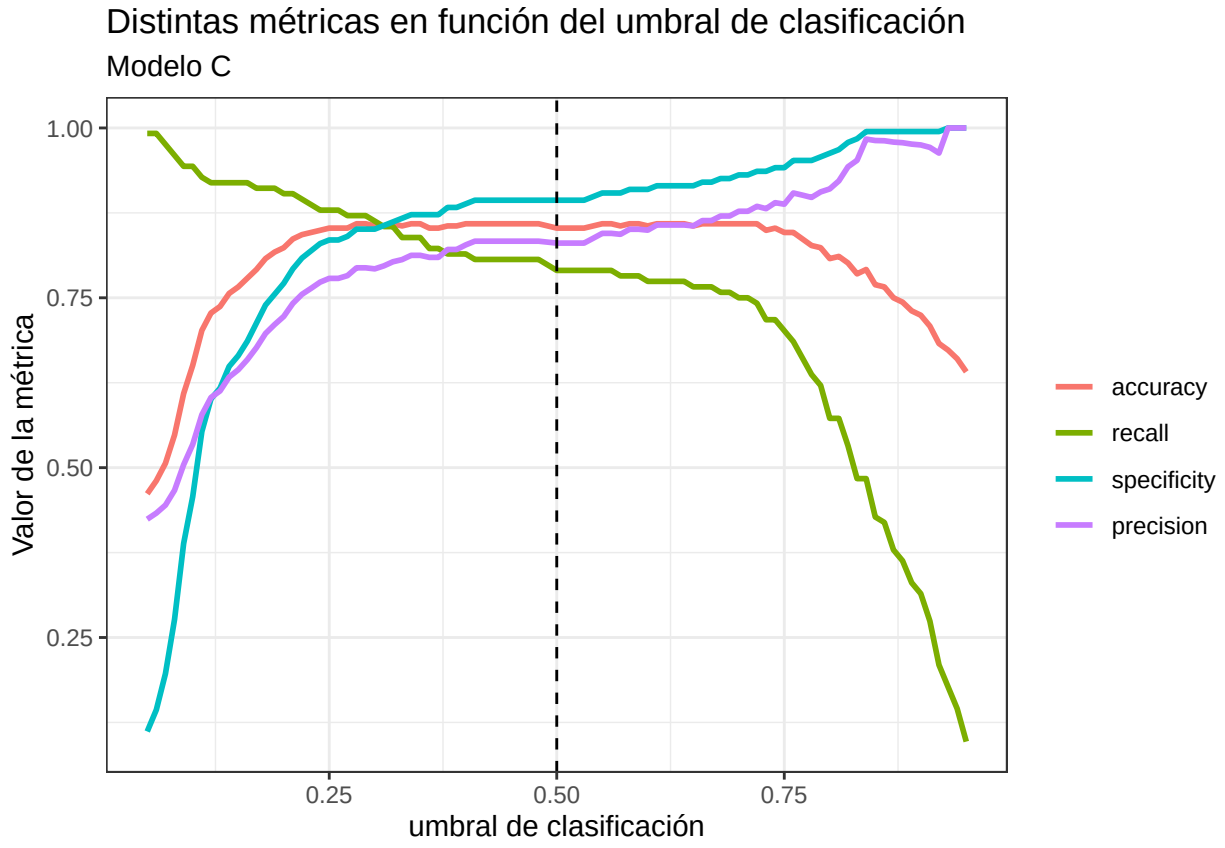
```r
datosV_m <- reshape2::melt(datosV,id.vars=c('umbral'))
colnames(datosV_m)[2] <- c('Metrica')
```

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.

```r
library(ggplot2)

u = 0.5 #Se dio un valor arbitrario, tú modificalo de acuerdo al criterio que selecciones.

ggplot(data=datosV_m, aes(x=umbral,y=value,color=Metrica)) + geom_line(size=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.
```



Define cuál es el mejor umbral en donde se obtienen las mejores métricas Recall, Accuracy, Sensitivity y Specificity.

El mejor umbral se puede encontrar en 0.50, donde todas las metricas se encuentran lo mas cercas las unas de las otras, en especial acurracy y precision, evitando un recall peor.

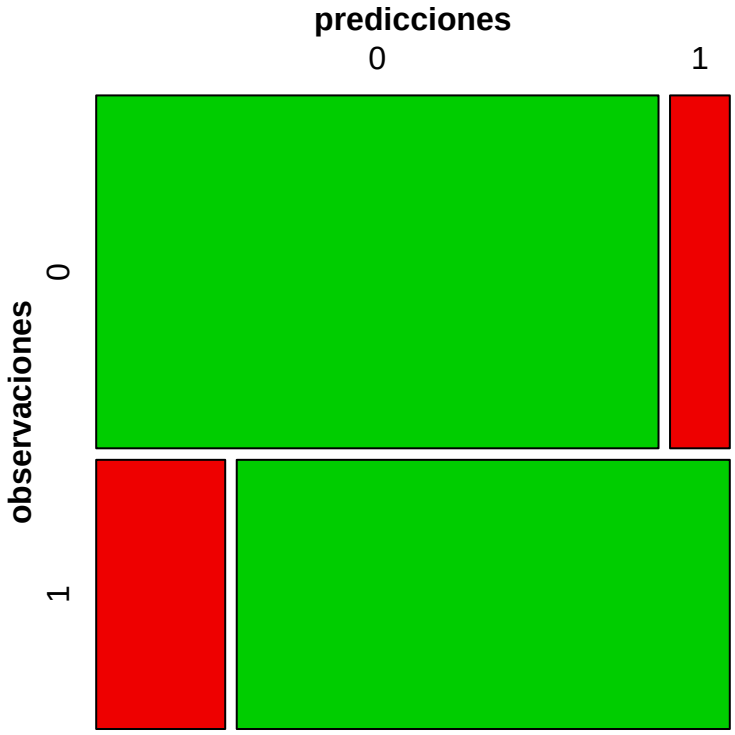
Matriz de confusión con el umbral de clasificación optimo

De acuerdo al umbral seleccionado, calcula la matriz de confusión y las métricas obtenidas. Indica si mejora la predicción con respecto al umbral de u = 0.5, que es el que se maneja por default.

```
prediccionesV = ifelse(pred_val > 0.3, yes = 1, no = 0)
M_Cv <- table(prediccionesV, M_valid$Survived, dnn = c("observaciones", "predicciones"))
M_Cv
```

observaciones/predicciones	0	1
0	160	17
1	28	107

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




```
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 0.8557692
```

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

```
## La Sensibilidad del modelo es 0.9039548
```

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

```
## La Especificidad del modelo es 0.7925926
```

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

```
## La Precisión del modelo es 0.8510638
```

Conclusiones

Concluye definiendo cuáles fueron las principales características de las personas que sobrevivieron e indica cuáles son los coeficientes de cada variable en el modelo de predicción de sobrevivencia.

Al analizar los datos, se pudo oservar que el sexo, la clase, la edad y el numero de familia fueron de gran imporotancia para la supervivencia de los pasajeros, esto se puede ver en como las mujeres tuvieron un mayor porcentaje de supervivencia en comparacion de los hombres, lo cual era algo mas comun en la epoca, por otra parte, se pudo oservar. como las clases mas privilegiadas eran las que teniann mayores probailidades de sobrevivir, a comparacion de las otras clases, a su vez, el tener un gran numero de familiares se tenia una mmenor probabilidad de supervivencia debido a la dificultad que habia en evacuar a grandes grupos de personas

Interpreta los coeficientes de predicción de cada variable. Indica cómo influyó en la sobrevivencia.

En el modelo seleccionado, el modelo c, se pudo ver como el intercepto (4.397) indica que, en ausencia de otras variables, la probabilidad base de supervivencia es alta, por otra parte, Pclass2 (-1.254) y Pclass3 (-2.152) nos dice que los pasajeros de la segunda clase tenian una menor probabilidad de supervivencia a comparacion de los de la clase alta, lo cual nos ayuda a resaltar la importancia de la clase social, por otra parte, en la variiable sexo, especiificamente a SexMale (-3.69) muestra que los hombres tenían muchas menos probabilidades de sobrevivir que las mujeres, a su vez, en cuanto la edad (-0.039) esta nos indica que a medida que aumentaba la edad, la probabilidad de sobrevivir disminuía levemente, finalmente, el coeficiente para SibSp (-0.295) muestra que las personas con mas familia tenían una probabilidad ligeramente menor de sobrevivir.

Indica cuál es el mejor umbral de clasificación y por qué.

Respecto al umbral de clasificación, se determina que se busca una mayor exactitud en los datos resultantes del modelo, por lo que se usa un umbral de 0.50 debido a que posee tanto como buena precision, como exactitud,auqnue no tan buen recall, la razon del por que es sencilla, ya que permite tener suficiente sensibilidad para poder determinar si la prediccion ese correcta, a la vez que se asegura que mantiene bajo control a los falsos positivos, por lo que es posible obtener los mejores resultados sin sacrificar las otras metricas.