# Actividad Integradora 2

Daniela Jiménez Téllez

2024-11-19

# Problema

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:

```
entrenamiento <- read.csv("Titanic.csv")

prueba <- read.csv("Titanic_test.csv")
```

# Instrucciones

**1. Prepara la base de datos Titanic:**

- Analiza los datos faltantes
- Realiza un análisis descriptivo
- 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.

```
# Datos faltantes

cat("---- DATOS FALTANTES ---- \n\n")
```

```
## ---- DATOS FALTANTES ----
```

```
sapply(entrenamiento, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0           0           0           0           0         263
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0           1           0           2
```

```
cat("\n\n")
```

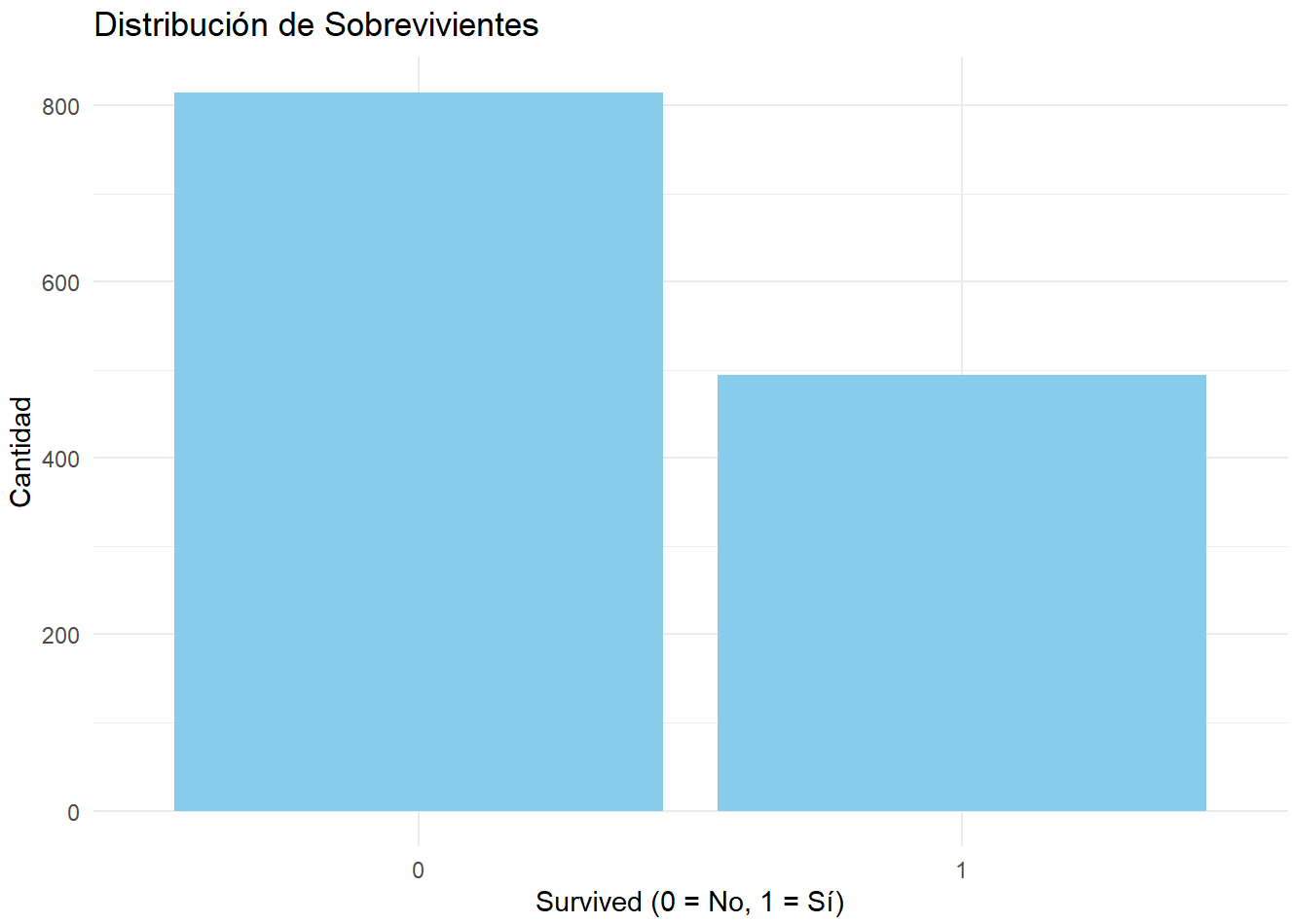
```
# Imputación de datos faltantes

entrenamiento$Age[is.na(entrenamiento$Age)] <- median(entrenamiento$Age, na.rm = TRUE)
entrenamiento$Embarked[is.na(entrenamiento$Embarked)] <- "S"

# Quitar columnas y hacer las categóricas a factores

entrenamiento <- entrenamiento %>%
  select(-c(Name, PassengerId, Ticket, Cabin)) %>%
  mutate(Sex = as.factor(Sex),
         Embarked = as.factor(Embarked),
         Pclass = as.factor(Pclass))

# Análisis descriptivo

cat("---- ANÁLISIS DESCRIPTIVO ---- \n\n")
```

```
## ---- ANÁLISIS DESCRIPTIVO ----
```

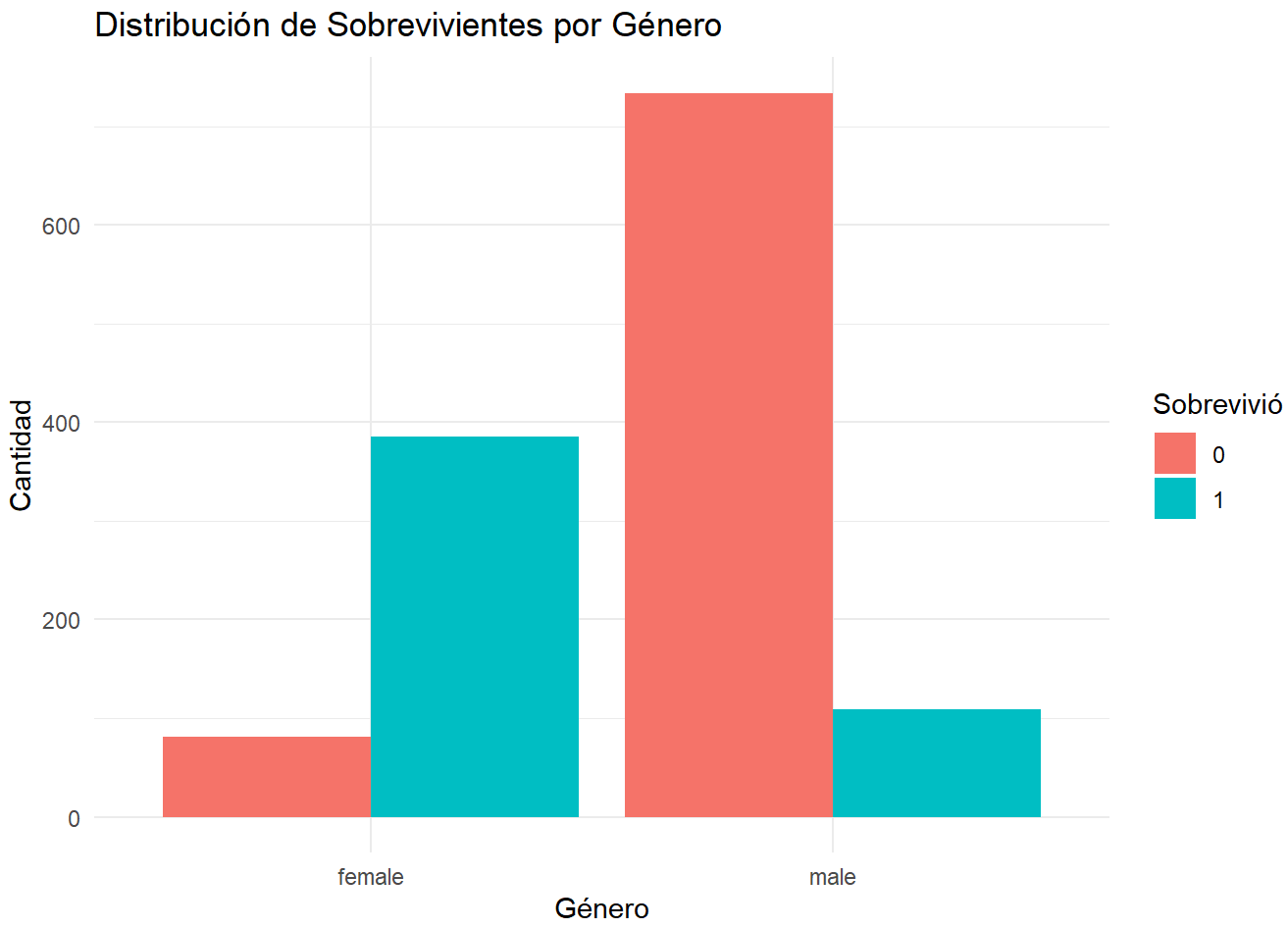
```
summary(entrenamiento)
```

```
##     Survived      Pclass      Sex           Age            SibSp
##  Min.   :0.0000   1:323   female:466   Min.   : 0.17   Min.   :0.0000
##  1st Qu.:0.0000   2:277   male  :843   1st Qu.:22.00   1st Qu.:0.0000
##  Median :0.0000   3:709                Median :28.00   Median :0.0000
##  Mean   :0.3774                        Mean   :29.50   Mean   :0.4989
##  3rd Qu.:1.0000                        3rd Qu.:35.00   3rd Qu.:1.0000
##  Max.   :1.0000                        Max.   :80.00   Max.   :8.0000
##
##      Parch            Fare         Embarked
##  Min.   :0.000   Min.   :  0.000   C:270
##  1st Qu.:0.000   1st Qu.:  7.896   Q:123
##  Median :0.000   Median : 14.454   S:916
##  Mean   :0.385   Mean   : 33.295
##  3rd Qu.:0.000   3rd Qu.: 31.275
##  Max.   :9.000   Max.   :512.329
##                  NA's   :1
```

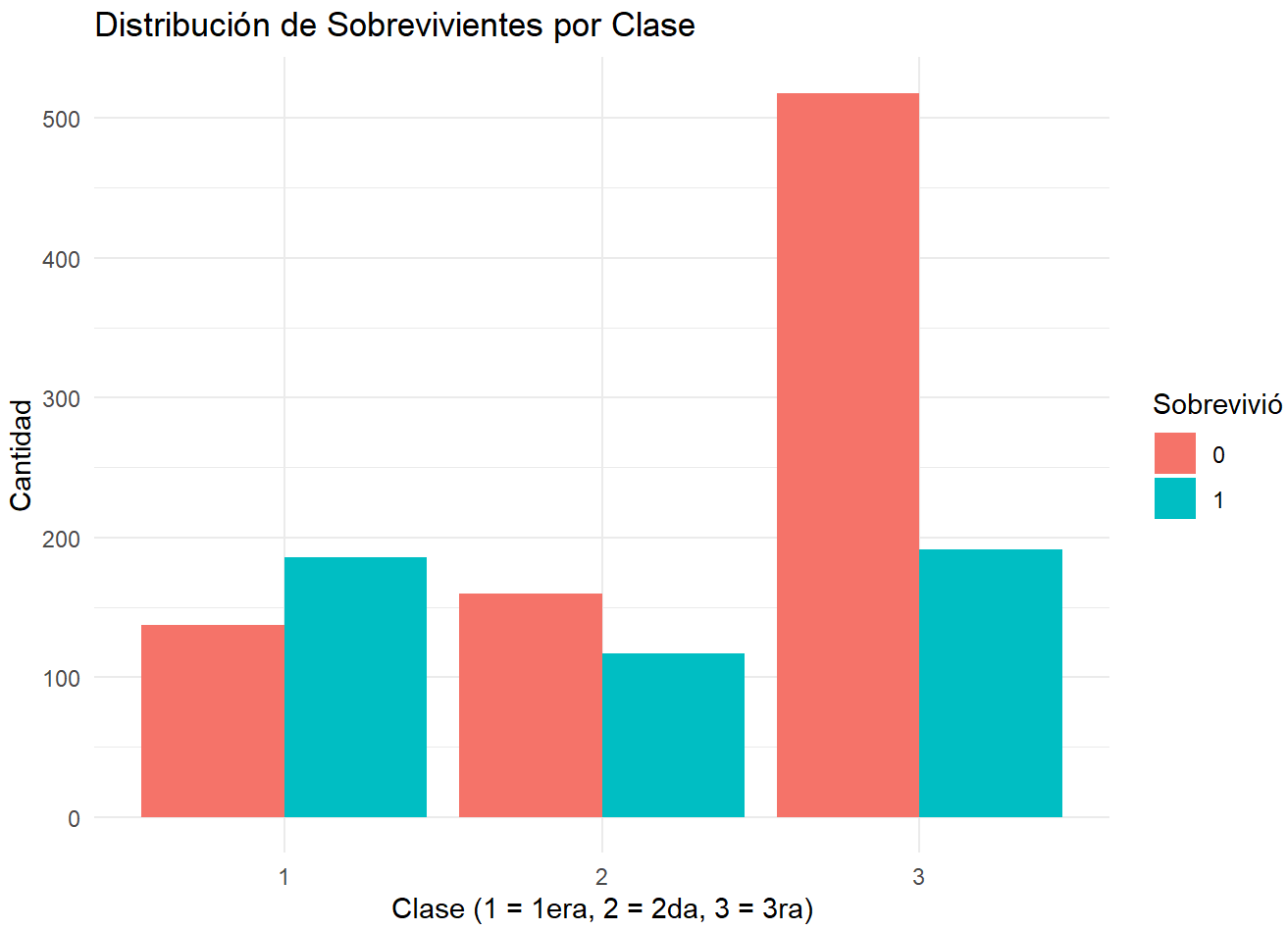
```
# Distribución de sobrevivientes

ggplot(entrenamiento, aes(x = as.factor(Survived))) +
  geom_bar(fill = "skyblue") +
  labs(title = "Distribución de Sobrevivientes",
       x = "Survived (0 = No, 1 = Sí)",
       y = "Cantidad") +
  theme_minimal()
```



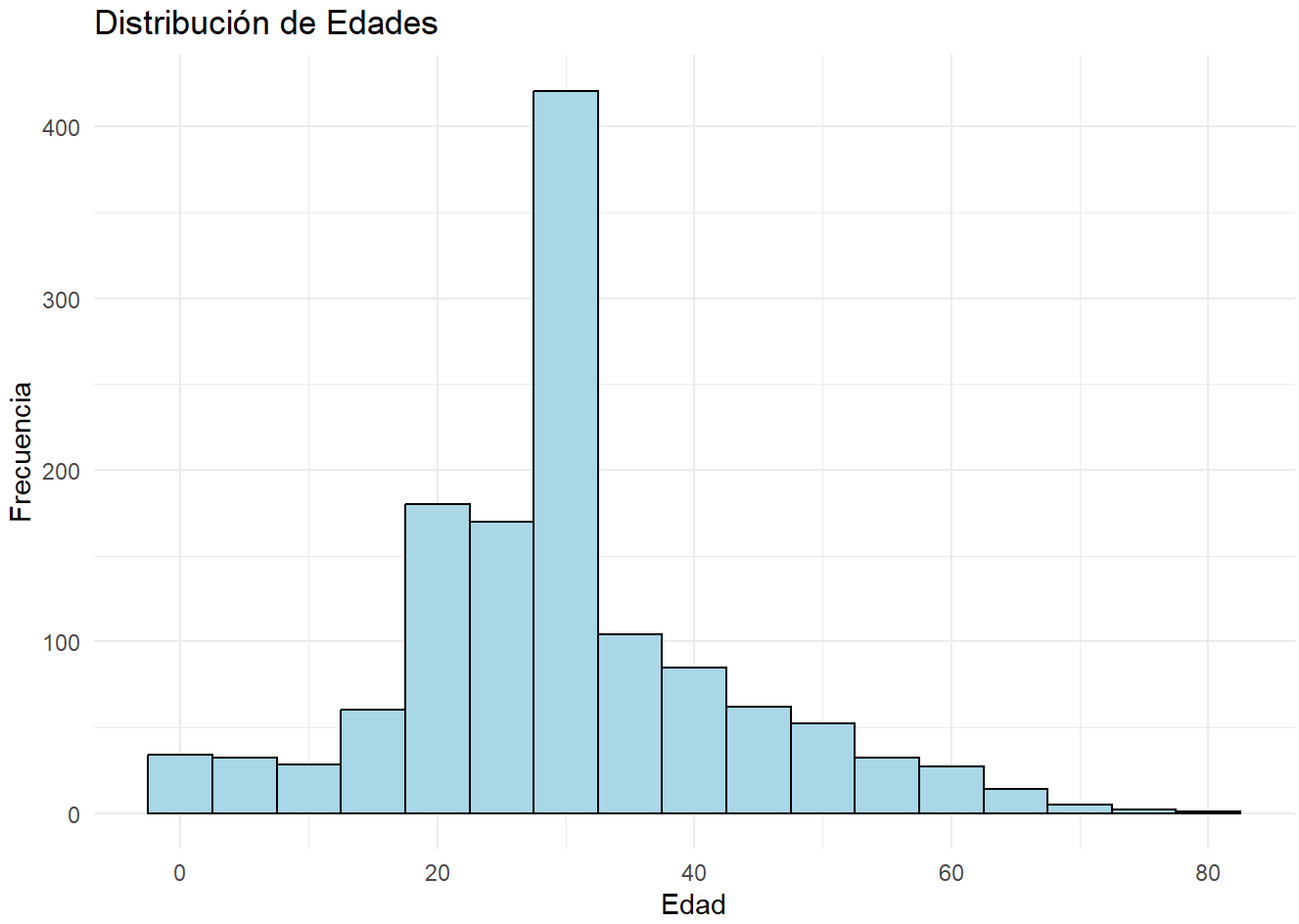

```
# Distribución por género

ggplot(entrenamiento, aes(x = Sex, fill = as.factor(Survived))) +
  geom_bar(position = "dodge") +
  labs(title = "Distribución de Sobrevivientes por Género",
       x = "Género",
       y = "Cantidad",
       fill = "Sobrevivió") +
  theme_minimal()
```



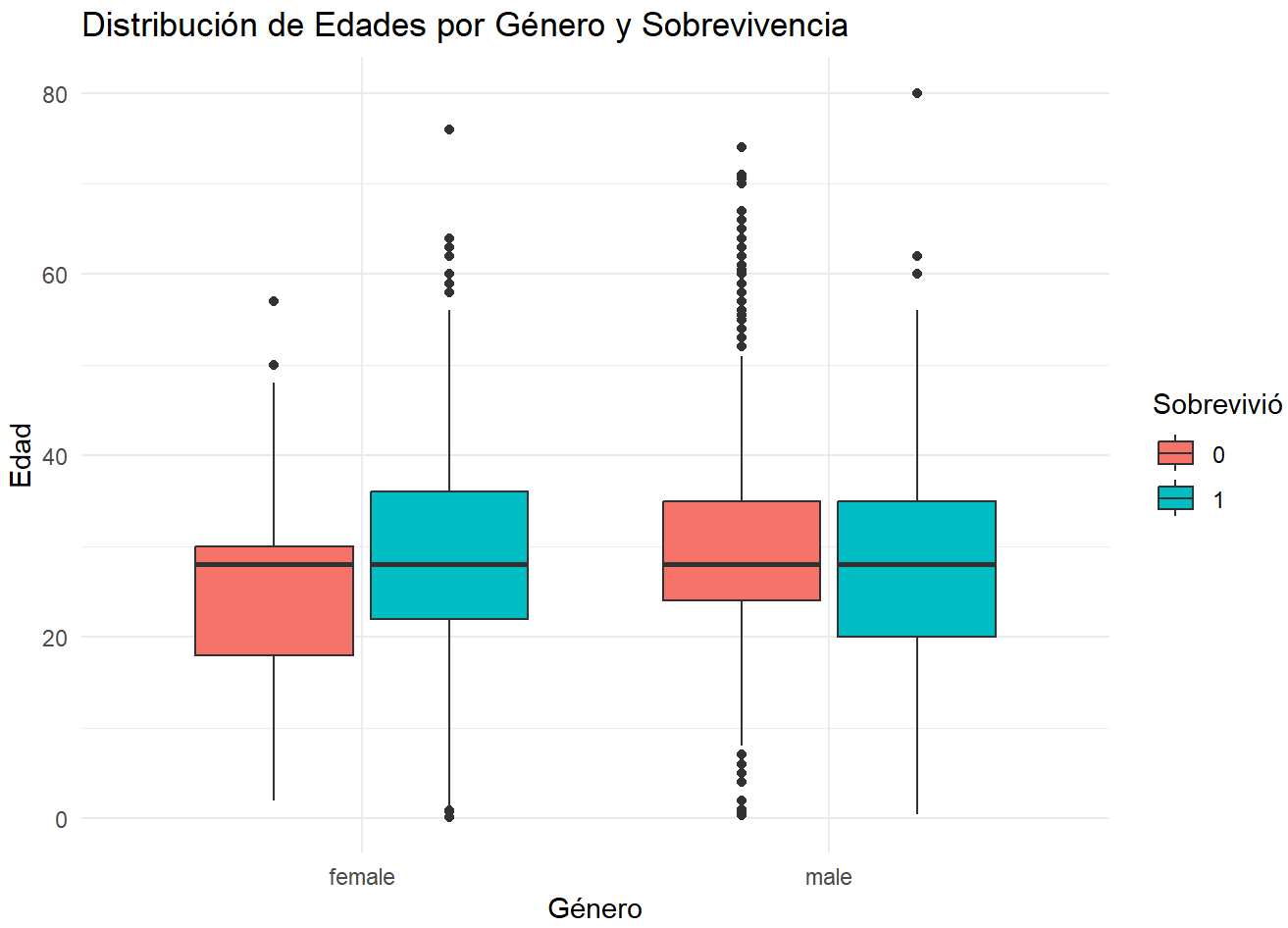

```
# Distribución por clase

ggplot(entrenamiento, aes(x = as.factor(Pclass), fill = as.factor(Survived))) +
  geom_bar(position = "dodge") +
  labs(title = "Distribución de Sobrevivientes por Clase",
       x = "Clase (1 = 1era, 2 = 2da, 3 = 3ra)",
       y = "Cantidad",
       fill = "Sobrevivió") +
  theme_minimal()
```




```
# Distribución de edades

ggplot(entrenamiento, aes(x = Age)) +
  geom_histogram(binwidth = 5, fill = "lightblue", color = "black") +
  labs(title = "Distribución de Edades",
       x = "Edad",
       y = "Frecuencia") +
  theme_minimal()
```

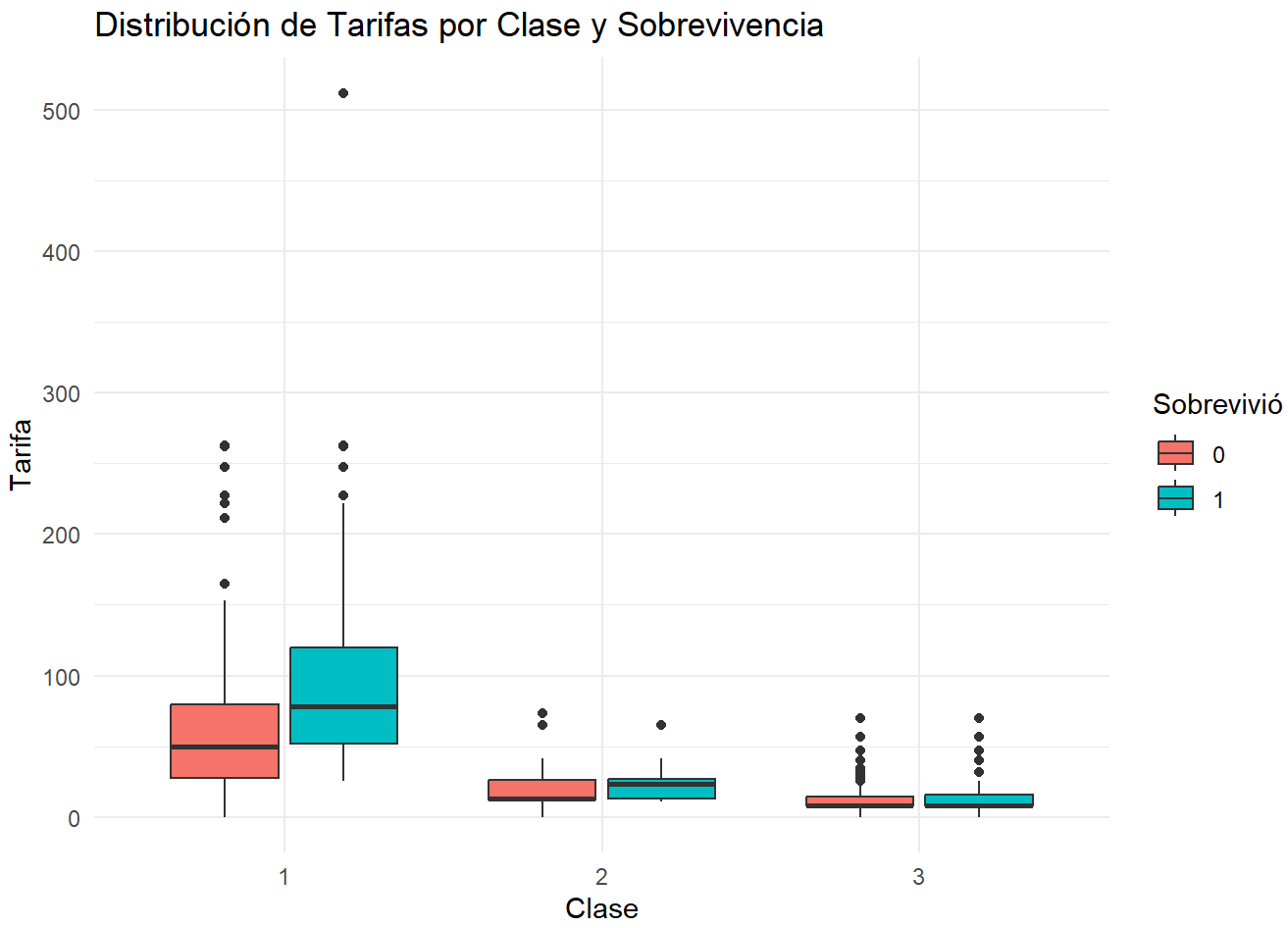



```
# Boxplot de edades por género y sobrevivencia

ggplot(entrenamiento, aes(x = Sex, y = Age, fill = as.factor(Survived))) +
  geom_boxplot() +
  labs(title = "Distribución de Edades por Género y Sobrevivencia",
       x = "Género",
       y = "Edad",
       fill = "Sobrevivió") +
  theme_minimal()
```

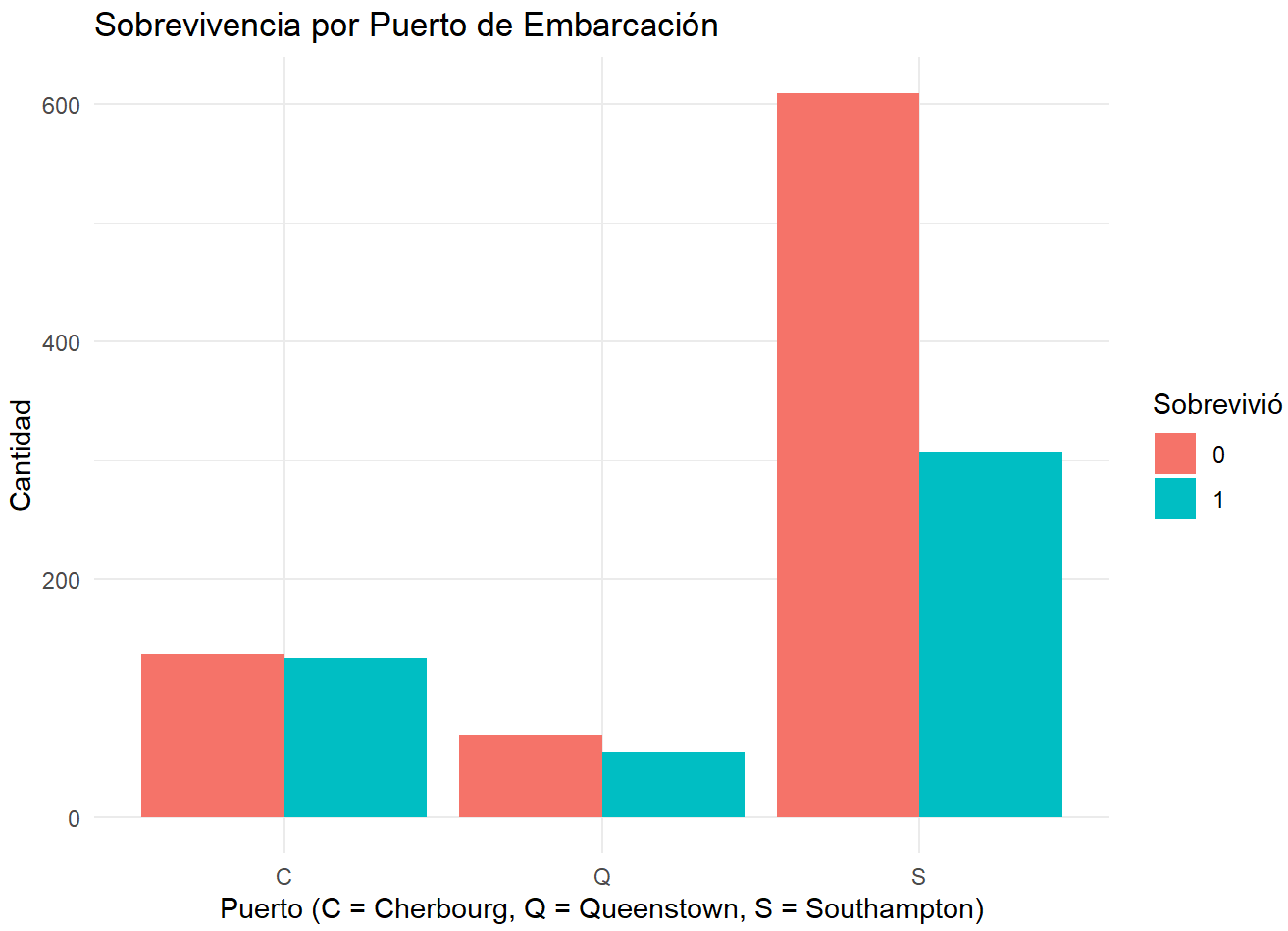



```
# Distribución de tarifa por clase y sobrevivencia

ggplot(entrenamiento, aes(x = as.factor(Pclass), y = Fare, fill = as.factor(Survived))) +
  geom_boxplot() +
  labs(title = "Distribución de Tarifas por Clase y Sobrevivencia",
       x = "Clase",
       y = "Tarifa",
       fill = "Sobrevivió") +
  theme_minimal()
```

```
## Warning: Removed 1 row containing non-finite outside the scale range
## (`stat_boxplot()`).
```




```
# Puerto de embarcación y sobrevivencia

ggplot(entrenamiento, aes(x = Embarked, fill = as.factor(Survived))) +
  geom_bar(position = "dodge") +
  labs(title = "Sobrevivencia por Puerto de Embarcación",
       x = "Puerto (C = Cherbourg, Q = Queenstown, S = Southampton)",
       y = "Cantidad",
       fill = "Sobrevivió") +
  theme_minimal()
```




```
# Partición de datos

cat("---- PARTICIÓN DE LOS DATOS ---- \n\n")
```

```
## ---- PARTICIÓN DE LOS DATOS ----
```

```
set.seed(123)
trainIndex <- createDataPartition(entrenamiento$Survived, p = 0.7, list = FALSE)
train <- entrenamiento[trainIndex, ]
validation <- entrenamiento[-trainIndex, ]

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

```
##
##         0         1
## 0.6183206 0.3816794
```

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

```
##
##         0         1
## 0.6326531 0.3673469
```

**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.

```
# Transformar variables categóricas a dummies

train <- dummy_cols(train,
                    select_columns = c("Sex", "Embarked", "Pclass"),
                    remove_first_dummy = TRUE,
                    remove_selected_columns = TRUE)

validation <- dummy_cols(validation,
                         select_columns = c("Sex", "Embarked", "Pclass"),
                         remove_first_dummy = TRUE,
                         remove_selected_columns = TRUE)

prueba <- dummy_cols(prueba,
                     select_columns = c("Sex", "Embarked", "Pclass"),
                     remove_first_dummy = TRUE,
                     remove_selected_columns = TRUE)

train <- na.omit(train)
validation <- na.omit(validation)
prueba <- na.omit(prueba)
```

```
# Modelo logístico completo

modelo_completo <- glm(Survived ~ ., data = train, family = binomial)
step(modelo_completo, direction = "both", trace = 1)
```

```
## Start:  AIC=696.88
## Survived ~ Age + SibSp + Parch + Fare + Sex_male + Embarked_Q +
##     Embarked_S + Pclass_2 + Pclass_3
##
##              Df Deviance     AIC
## - Embarked_S  1   676.93  694.93
## - Parch       1   677.21  695.21
## - Embarked_Q  1   677.33  695.33
## - Fare        1   678.85  696.85
## <none>            676.88  696.88
## - SibSp       1   685.50  703.50
## - Pclass_2    1   692.11  710.11
## - Age         1   699.80  717.80
## - Pclass_3    1   724.94  742.94
## - Sex_male    1  1059.01 1077.01
##
## Step:  AIC=694.93
## Survived ~ Age + SibSp + Parch + Fare + Sex_male + Embarked_Q +
##     Pclass_2 + Pclass_3
##
##              Df Deviance     AIC
## - Parch       1   677.27  693.27
## - Embarked_Q  1   677.75  693.75
## <none>            676.93  694.93
## - Fare        1   679.11  695.11
## + Embarked_S  1   676.88  696.88
## - SibSp       1   685.81  701.81
## - Pclass_2    1   692.93  708.93
## - Age         1   700.02  716.02
## - Pclass_3    1   725.93  741.93
## - Sex_male    1  1060.12 1076.12
##
## Step:  AIC=693.27
## Survived ~ Age + SibSp + Fare + Sex_male + Embarked_Q + Pclass_2 +
##     Pclass_3
##
##              Df Deviance     AIC
## - Embarked_Q  1   678.27  692.27
## - Fare        1   679.21  693.21
## <none>            677.27  693.27
## + Parch       1   676.93  694.93
## + Embarked_S  1   677.21  695.21
## - SibSp       1   687.69  701.69
## - Pclass_2    1   693.88  707.88
## - Age         1   700.35  714.35
## - Pclass_3    1   728.63  742.63
## - Sex_male    1  1071.02 1085.02
##
## Step:  AIC=692.27
## Survived ~ Age + SibSp + Fare + Sex_male + Pclass_2 + Pclass_3
##
##              Df Deviance     AIC
```

```
## - Fare        1   680.18  692.18
## <none>            678.27  692.27
## + Embarked_Q  1   677.27  693.27
## + Embarked_S  1   677.73  693.73
## + Parch       1   677.75  693.75
## - SibSp       1   689.90  701.90
## - Pclass_2    1   694.77  706.77
## - Age         1   700.81  712.81
## - Pclass_3    1   728.70  740.70
## - Sex_male    1  1085.08 1097.08
##
## Step:  AIC=692.18
## Survived ~ Age + SibSp + Sex_male + Pclass_2 + Pclass_3
##
##              Df Deviance     AIC
## <none>            680.18  692.18
## + Fare        1   678.27  692.27
## + Embarked_Q  1   679.21  693.21
## + Embarked_S  1   679.29  693.29
## + Parch       1   679.98  693.98
## - SibSp       1   690.47  700.47
## - Age         1   703.58  713.58
## - Pclass_2    1   707.48  717.48
## - Pclass_3    1   774.13  784.13
## - Sex_male    1  1100.04 1110.04
```

```
##
## Call:  glm(formula = Survived ~ Age + SibSp + Sex_male + Pclass_2 +
##     Pclass_3, family = binomial, data = train)
##
## Coefficients:
## (Intercept)          Age        SibSp     Sex_male     Pclass_2     Pclass_3
##     4.58822     -0.04037     -0.29437     -3.65933     -1.49436     -2.45874
##
## Degrees of Freedom: 915 Total (i.e. Null);  910 Residual
## Null Deviance:       1218
## Residual Deviance: 680.2     AIC: 692.2
```

```
summary(modelo_completo)
```

```
##
## Call:
## glm(formula = Survived ~ ., family = binomial, data = train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.355973   0.510988   8.525  < 2e-16 ***
## Age         -0.040149   0.008681  -4.625 3.74e-06 ***
## SibSp       -0.286834   0.103212  -2.779 0.005451 **
## Parch       -0.063380   0.110660  -0.573 0.566817
## Fare         0.003204   0.002382   1.345 0.178484
## Sex_male    -3.633589   0.227030 -16.005  < 2e-16 ***
## Embarked_Q   0.277019   0.413585   0.670 0.502986
## Embarked_S  -0.056552   0.263847  -0.214 0.830284
## Pclass_2    -1.281809   0.330417  -3.879 0.000105 ***
## Pclass_3    -2.272608   0.326277  -6.965 3.28e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1218.43  on 915  degrees of freedom
## Residual deviance:  676.88  on 906  degrees of freedom
## AIC: 696.88
##
## Number of Fisher Scoring iterations: 5
```

```
# Modelo logístico con algunas variables

variables <- Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age + SibSp + Fare + Embarked_S + Embar
ked_Q

modelo_reducido <- glm(variables, data = train, family = binomial)
step(modelo_reducido, direction = "both", trace = 1)
```

```
## Start:  AIC=695.21
## Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age + SibSp + Fare +
##     Embarked_S + Embarked_Q
##
##              Df Deviance     AIC
## - Embarked_S  1   677.27  693.27
## - Embarked_Q  1   677.73  693.73
## - Fare        1   678.95  694.95
## <none>            677.21  695.21
## - SibSp       1   687.23  703.23
## - Pclass_2    1   692.91  708.91
## - Age         1   700.09  716.09
## - Pclass_3    1   727.33  743.33
## - Sex_male    1  1070.26 1086.26
##
## Step:  AIC=693.27
## Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age + SibSp + Fare +
##     Embarked_Q
##
##              Df Deviance     AIC
## - Embarked_Q  1   678.27  692.27
## - Fare        1   679.21  693.21
## <none>            677.27  693.27
## + Embarked_S  1   677.21  695.21
## - SibSp       1   687.69  701.69
## - Pclass_2    1   693.88  707.88
## - Age         1   700.35  714.35
## - Pclass_3    1   728.63  742.63
## - Sex_male    1  1071.02 1085.02
##
## Step:  AIC=692.27
## Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age + SibSp + Fare
##
##              Df Deviance     AIC
## - Fare        1   680.18  692.18
## <none>            678.27  692.27
## + Embarked_Q  1   677.27  693.27
## + Embarked_S  1   677.73  693.73
## - SibSp       1   689.90  701.90
## - Pclass_2    1   694.77  706.77
## - Age         1   700.81  712.81
## - Pclass_3    1   728.70  740.70
## - Sex_male    1  1085.08 1097.08
##
## Step:  AIC=692.18
## Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age + SibSp
##
##              Df Deviance     AIC
## <none>            680.18  692.18
## + Fare        1   678.27  692.27
## + Embarked_Q  1   679.21  693.21
## + Embarked_S  1   679.29  693.29
```

```
## - SibSp       1   690.47  700.47
## - Age         1   703.58  713.58
## - Pclass_2    1   707.48  717.48
## - Pclass_3    1   774.13  784.13
## - Sex_male    1  1100.04 1110.04
```

```
##
## Call:  glm(formula = Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age +
##     SibSp, family = binomial, data = train)
##
## Coefficients:
## (Intercept)     Pclass_2     Pclass_3     Sex_male          Age        SibSp
##     4.58822     -1.49436     -2.45874     -3.65933     -0.04037     -0.29437
##
## Degrees of Freedom: 915 Total (i.e. Null);  910 Residual
## Null Deviance:       1218
## Residual Deviance: 680.2     AIC: 692.2
```

```
summary(modelo_reducido)
```

```
##
## Call:
## glm(formula = variables, family = binomial, data = train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.345659   0.509893   8.523  < 2e-16 ***
## Pclass_2    -1.295762   0.329315  -3.935 8.33e-05 ***
## Pclass_3    -2.294274   0.323905  -7.083 1.41e-12 ***
## Sex_male    -3.607767   0.221762 -16.269  < 2e-16 ***
## Age         -0.040105   0.008678  -4.622 3.81e-06 ***
## SibSp       -0.300179   0.100857  -2.976  0.00292 **
## Fare         0.002935   0.002310   1.271  0.20387
## Embarked_S  -0.066003   0.262841  -0.251  0.80172
## Embarked_Q   0.296410   0.411200   0.721  0.47101
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1218.43  on 915  degrees of freedom
## Residual deviance:  677.21  on 907  degrees of freedom
## AIC: 695.21
##
## Number of Fisher Scoring iterations: 5
```

```
# Comparar AIC entre modelos

aic_completo <- AIC(modelo_completo)
aic_reducido <- AIC(modelo_reducido)

cat("AIC Modelo Completo:", aic_completo, "\n")
```

```
## AIC Modelo Completo: 696.8818
```

```
cat("AIC Modelo Reducido:", aic_reducido, "\n")
```

```
## AIC Modelo Reducido: 695.209
```

Se puede observar que el AIC del modelo completo es más grande, lo que nos dice que el modelo reducido muestra mejores resultados.

**3. Analiza los modelos a través de:**

- Identificación de la Desviación residual de cada modelo
- Identificación de la Desviación nula
- Cálculo de la Desviación Explicada
- Prueba de la razón de verosimilitud
- Define cuál es el mejor modelo
- Escribe su ecuación, analiza sus coeficientes y detecta el efecto de cada predictor en la clasificación.

```
# Desviación residual y nula para el modelo completo

desviacion_residual_completo <- modelo_completo$deviance
desviacion_nula_completo <- modelo_completo$null.deviance

cat("Desviación Residual Modelo Completo:", desviacion_residual_completo, "\n")
```

```
## Desviación Residual Modelo Completo: 676.8818
```

```
cat("Desviación Nula Modelo Completo:", desviacion_nula_completo, "\n")
```

```
## Desviación Nula Modelo Completo: 1218.428
```

```
# Desviación residual y nula para el modelo reducido

desviacion_residual_reducido <- modelo_reducido$deviance
desviacion_nula_reducido <- modelo_reducido$null.deviance

cat("Desviación Residual Modelo Reducido:", desviacion_residual_reducido, "\n")
```

```
## Desviación Residual Modelo Reducido: 677.209
```

```
cat("Desviación Nula Modelo Reducido:", desviacion_nula_reducido, "\n")
```

```
## Desviación Nula Modelo Reducido: 1218.428
```

```
# Desviación explicada para el modelo completo

desviacion_explicada_completo <- 1 - (desviacion_residual_completo / desviacion_nula_completo)
cat("Desviación Explicada Modelo Completo:", desviacion_explicada_completo, "\n")
```

```
## Desviación Explicada Modelo Completo: 0.4444631
```

```
# Desviación explicada para el modelo reducido

desviacion_explicada_reducido <- 1 - (desviacion_residual_reducido / desviacion_nula_reducido)
cat("Desviación Explicada Modelo Reducido:", desviacion_explicada_reducido, "\n\n")
```

```
## Desviación Explicada Modelo Reducido: 0.4441946
```

```
# Prueba de razón de verosimilitud

prueba_lrt <- anova(modelo_reducido, modelo_completo, test = "Chisq")

cat("Prueba de Razón de Verosimilitud: \n")
```

```
## Prueba de Razón de Verosimilitud:
```

```
print(prueba_lrt)
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ Pclass_2 + Pclass_3 + Sex_male + Age + SibSp + Fare +
##     Embarked_S + Embarked_Q
## Model 2: Survived ~ Age + SibSp + Parch + Fare + Sex_male + Embarked_Q +
##     Embarked_S + Pclass_2 + Pclass_3
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       907     677.21
## 2       906     676.88  1  0.32721   0.5673
```

El modelo reducido es mejor porque logra un buen equilibrio entre ser simple y funcionar bien. Aunque el modelo completo tiene un ajuste un poco mejor, la diferencia es tan pequeña que no es significativa en la prueba de verosimilitud. Además, el modelo reducido tiene un AIC más bajo, lo que indica que es más eficiente al no incluir variables que no aportan mucho al resultado. Ambos modelos explican prácticamente la misma cantidad de la variabilidad, pero el reducido es más fácil de interpretar y probablemente se generalice mejor a otros datos.

**4. Analiza las predicciones para los datos de entrenamiento**

- Elabora la matriz de confusión
- Elabora la Curva ROC
- Elabora el gráfico de violín
- Concluye sobre el modelo basándote en las predicciones de los datos de entrenamiento.

```
# Predicciones en datos de entrenamiento

predicciones <- ifelse(predict(modelo_reducido, type = "response") > 0.5, 1, 0)

# Matriz de confusión

confusion_matrix <- table(Real = train$Survived, Predicho = predicciones)
print(confusion_matrix)
```

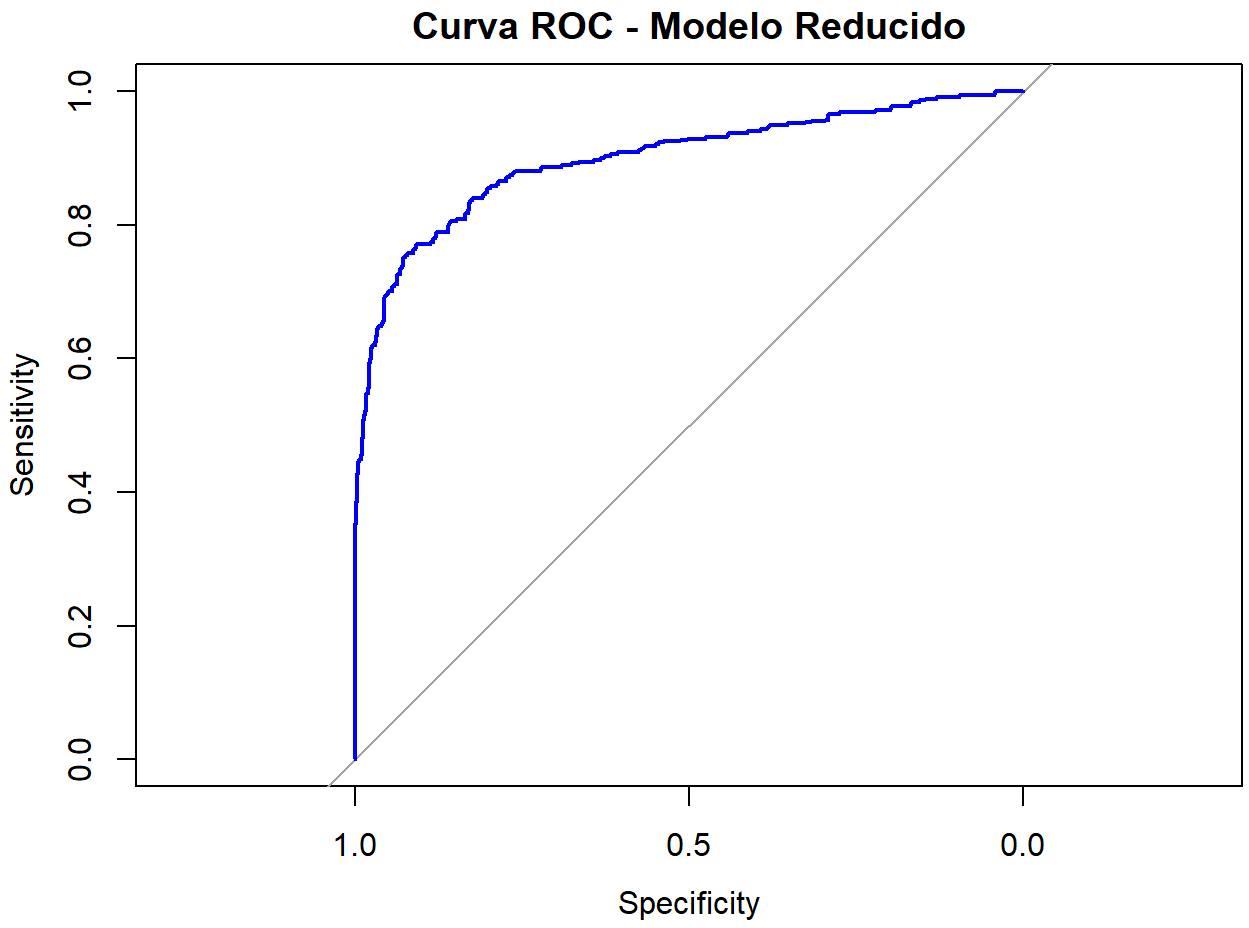
```
##     Predicho
## Real   0   1
##    0 502  64
##    1  79 271
```

```
# Curva ROC

roc_obj <- roc(train$Survived, predict(modelo_reducido, type = "response"))
```

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

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

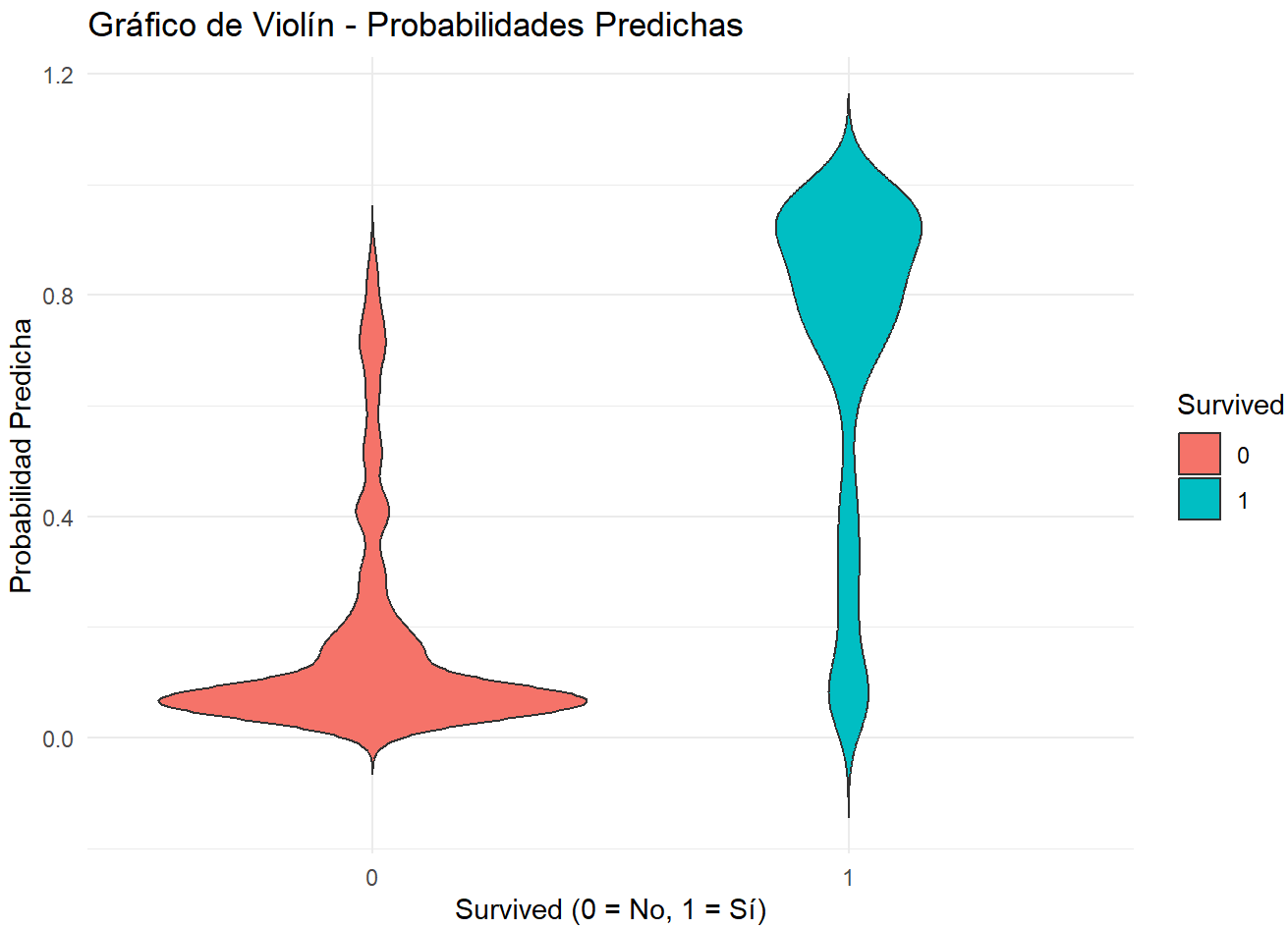
```
plot(roc_obj, col = "blue", main = "Curva ROC - Modelo Reducido")
```




```
auc_value <- auc(roc_obj)
print(paste("Área bajo la curva (AUC):", auc_value))
```

```
## [1] "Área bajo la curva (AUC): 0.898672387682988"
```

```
# Crear un dataframe con las predicciones y los valores reales

data_plot <- data.frame(
  Predicciones = predict(modelo_reducido, type = "response"),
  Survived = as.factor(train$Survived))

# Gráfico de violín

ggplot(data_plot, aes(x = Survived, y = Predicciones, fill = Survived)) +
  geom_violin(trim = FALSE) +
  labs(title = "Gráfico de Violín - Probabilidades Predichas",
       x = "Survived (0 = No, 1 = Sí)",
       y = "Probabilidad Predicha") +
  theme_minimal()
```




El modelo reducido tiene un buen desempeño para ver si alguien sobrevive o no en el Titanic. El gráfico de violín demuestra una separación entre las probabilidades predichas de sobrevivientes y no sobrevivientes. La curva ROC, con un AUC de 0.898, muestra buenos resultados. La matriz de confusión muestra que el modelo clasifica correctamente a la mayoría de los casos, aunque existen algunos falsos positivos (64) y falsos negativos (79). En general, el modelo es confiable para predecir la supervivencia con precisión y balance entre sensibilidad y especificidad.

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

- Elije un umbral de clasificación óptimo
- Elabora la matriz de confusión con el umbral de clasificación óptimo

```
validation <- na.omit(validation)

# Predicciones

predicciones_val <- predict(modelo_reducido, newdata = validation, type = "response")

# Clasificación

umbral_optimo <- coords(roc(validation$Survived, predicciones_val), "best", ret = "threshold")
[[1]]
```

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

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

```
cat("Umbral óptimo:", umbral_optimo, "\n")
```

```
## Umbral óptimo: 0.5721826
```

```
predicciones_clasificadas <- ifelse(predicciones_val > umbral_optimo, 1, 0)

# Matriz de confusión

confusion_matrix_val <- table(Real = validation$Survived, Predicho = predicciones_clasificadas)
print(confusion_matrix_val)
```

```
##     Predicho
## Real   0   1
##    0 228  20
##    1  30 114
```

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

```
prueba <- na.omit(prueba)

predicciones_prueba <- predict(modelo_reducido, newdata = prueba, type = "response")
predicciones_clasificadas_prueba <- ifelse(predicciones_prueba > umbral_optimo, 1, 0)

resultados_prueba <- data.frame(
  PassengerId = prueba$PassengerId,
  Survived = predicciones_clasificadas_prueba)

resultados_prueba
```

```
##     PassengerId Survived
## 1           892        0
## 2           893        0
## 3           894        0
## 4           895        0
## 5           896        1
## 6           897        0
## 7           898        1
## 8           899        0
## 9           900        1
## 10          901        0
## 12          903        0
## 13          904        1
## 14          905        0
## 15          906        1
## 16          907        1
## 17          908        0
## 18          909        0
## 19          910        1
## 20          911        0
## 21          912        0
## 22          913        0
## 24          915        0
## 25          916        1
## 26          917        0
## 27          918        1
## 28          919        0
## 29          920        0
## 31          922        0
## 32          923        0
## 33          924        1
## 35          926        0
## 36          927        0
## 38          929        1
## 39          930        0
## 41          932        0
## 43          934        0
## 44          935        1
## 45          936        1
## 46          937        0
## 47          938        0
## 49          940        1
## 50          941        1
## 51          942        0
## 52          943        0
## 53          944        1
## 54          945        1
## 56          947        0
## 57          948        0
## 58          949        0
## 60          951        1
## 61          952        0
```

```
## 62          953        0
## 63          954        0
## 64          955        1
## 65          956        1
## 67          958        1
## 68          959        0
## 69          960        0
## 70          961        1
## 71          962        1
## 72          963        0
## 73          964        1
## 74          965        0
## 75          966        1
## 76          967        0
## 78          969        1
## 79          970        0
## 80          971        1
## 81          972        0
## 82          973        0
## 83          974        0
## 87          978        1
## 88          979        1
## 90          981        0
## 91          982        1
## 93          984        1
## 95          986        0
## 96          987        0
## 97          988        1
## 98          989        0
## 99          990        1
## 100         991        0
## 101         992        1
## 102         993        0
## 104         995        0
## 105         996        1
## 106         997        0
## 107         998        0
## 110        1001        0
## 111        1002        0
## 113        1004        1
## 114        1005        1
## 115        1006        1
## 116        1007        0
## 118        1009        1
## 119        1010        0
## 120        1011        1
## 121        1012        1
## 123        1014        1
## 124        1015        0
## 126        1017        1
## 127        1018        0
## 129        1020        0
```

```
## 130        1021        0
## 131        1022        0
## 132        1023        0
## 135        1026        0
## 136        1027        0
## 137        1028        0
## 138        1029        0
## 139        1030        1
## 140        1031        0
## 141        1032        0
## 142        1033        1
## 143        1034        0
## 144        1035        0
## 145        1036        0
## 146        1037        0
## 148        1039        0
## 150        1041        0
## 151        1042        1
## 154        1045        1
## 155        1046        0
## 156        1047        0
## 157        1048        1
## 158        1049        1
## 159        1050        0
## 160        1051        1
## 162        1053        0
## 163        1054        1
## 165        1056        0
## 166        1057        1
## 167        1058        0
## 168        1059        0
## 170        1061        1
## 172        1063        0
## 173        1064        0
## 175        1066        0
## 176        1067        1
## 177        1068        1
## 178        1069        0
## 179        1070        1
## 180        1071        1
## 181        1072        0
## 182        1073        0
## 183        1074        1
## 185        1076        1
## 186        1077        0
## 187        1078        1
## 188        1079        0
## 190        1081        0
## 191        1082        0
## 193        1084        0
## 194        1085        0
## 195        1086        0
```

```
## 196        1087        0
## 197        1088        1
## 198        1089        1
## 199        1090        0
## 202        1093        0
## 203        1094        0
## 204        1095        1
## 205        1096        0
## 207        1098        1
## 208        1099        0
## 209        1100        1
## 210        1101        0
## 211        1102        0
## 213        1104        0
## 214        1105        1
## 215        1106        0
## 216        1107        0
## 218        1109        0
## 219        1110        1
## 221        1112        1
## 222        1113        0
## 223        1114        1
## 224        1115        0
## 225        1116        1
## 227        1118        0
## 229        1120        0
## 230        1121        0
## 231        1122        0
## 232        1123        1
## 233        1124        0
## 235        1126        0
## 236        1127        0
## 237        1128        0
## 238        1129        0
## 239        1130        1
## 240        1131        1
## 241        1132        1
## 242        1133        1
## 243        1134        0
## 246        1137        0
## 247        1138        1
## 248        1139        0
## 249        1140        1
## 251        1142        1
## 252        1143        0
## 253        1144        0
## 254        1145        0
## 255        1146        0
## 258        1149        0
## 259        1150        1
## 260        1151        0
## 261        1152        0
```

```
## 262        1153        0
## 263        1154        1
## 264        1155        1
## 265        1156        0
## 270        1161        0
## 271        1162        0
## 273        1164        1
## 276        1167        1
## 277        1168        0
## 278        1169        0
## 279        1170        0
## 280        1171        0
## 281        1172        1
## 282        1173        0
## 284        1175        1
## 285        1176        1
## 286        1177        0
## 288        1179        0
## 292        1183        1
## 294        1185        0
## 295        1186        0
## 296        1187        0
## 297        1188        1
## 299        1190        0
## 300        1191        0
## 301        1192        0
## 303        1194        0
## 304        1195        0
## 306        1197        1
## 307        1198        0
## 308        1199        0
## 309        1200        0
## 310        1201        0
## 311        1202        0
## 312        1203        0
## 314        1205        1
## 315        1206        1
## 316        1207        1
## 317        1208        0
## 318        1209        0
## 319        1210        0
## 320        1211        0
## 321        1212        0
## 322        1213        0
## 323        1214        0
## 324        1215        0
## 325        1216        1
## 326        1217        0
## 327        1218        1
## 328        1219        0
## 329        1220        0
## 330        1221        0
```

```
## 331        1222        1
## 332        1223        0
## 334        1225        1
## 335        1226        0
## 336        1227        0
## 337        1228        0
## 338        1229        0
## 339        1230        0
## 341        1232        0
## 342        1233        0
## 344        1235        1
## 346        1237        1
## 347        1238        0
## 348        1239        1
## 349        1240        0
## 350        1241        1
## 351        1242        1
## 352        1243        0
## 353        1244        0
## 354        1245        0
## 355        1246        1
## 356        1247        0
## 357        1248        1
## 360        1251        1
## 361        1252        0
## 362        1253        1
## 363        1254        1
## 364        1255        0
## 365        1256        1
## 368        1259        1
## 369        1260        1
## 370        1261        0
## 371        1262        0
## 372        1263        1
## 373        1264        0
## 374        1265        0
## 375        1266        1
## 376        1267        1
## 377        1268        1
## 378        1269        0
## 379        1270        0
## 380        1271        0
## 382        1273        0
## 384        1275        1
## 386        1277        1
## 387        1278        0
## 388        1279        0
## 389        1280        0
## 390        1281        0
## 391        1282        0
## 392        1283        1
## 393        1284        0
```

```
## 394        1285        0
## 395        1286        0
## 396        1287        1
## 397        1288        0
## 398        1289        1
## 399        1290        0
## 400        1291        0
## 401        1292        1
## 402        1293        0
## 403        1294        1
## 404        1295        0
## 405        1296        0
## 406        1297        0
## 407        1298        0
## 408        1299        0
## 410        1301        1
## 412        1303        1
## 413        1304        1
## 415        1306        1
## 416        1307        0
```

**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?
- Interpreta los coeficientes del modelo
- Define cuál es el mejor umbral de clasificación y por qué

```
# Coeficientes

coeficientes <- summary(modelo_reducido)$coefficients
print(coeficientes)
```

```
##                Estimate  Std. Error     z value     Pr(>|z|)
## (Intercept)  4.345659220 0.509892685   8.5226938 1.558845e-17
## Pclass_2    -1.295762239 0.329314969  -3.9347201 8.329369e-05
## Pclass_3    -2.294274421 0.323905491  -7.0831600 1.409038e-12
## Sex_male    -3.607766730 0.221761787 -16.2686582 1.647279e-59
## Age         -0.040105433 0.008677716  -4.6216579 3.806854e-06
## SibSp       -0.300179203 0.100856690  -2.9762944 2.917546e-03
## Fare         0.002934529 0.002309562   1.2705994 2.038712e-01
## Embarked_S  -0.066003486 0.262841169  -0.2511155 8.017248e-01
## Embarked_Q   0.296410145 0.411199509   0.7208427 4.710063e-01
```

```
# Umbral

roc_val <- roc(validation$Survived, predict(modelo_reducido, newdata = validation, type = "respo
nse"))
```

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

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

```
umbral_optimo <- coords(roc_val, "best", ret = "threshold")[[1]]

cat("El umbral óptimo de clasificación es:", umbral_optimo, "\n")
```

```
## El umbral óptimo de clasificación es: 0.5721826
```

El análisis muestra que las principales características que influyen en la probabilidad de sobrevivir en el Titanic son el género, la clase, la tarifa, la edad y el puerto de embarque. Las mujeres, los pasajeros en primera clase, los que pagaron tarifas más altas y las personas más jóvenes tuvieron mayores probabilidades de sobrevivir. En el modelo, ser hombre o estar en tercera clase disminuye significativamente las probabilidades, mientras que una tarifa más alta incrementa la posibilidad de sobrevivir.