Titanic Lab

Javier Diaz & Juan Salamanca

2016

```
# Librerias:
library(ggplot2)
library(lattice)
library(caret)
library(e1071)
library(knitr)
```

Dataset: Titanic, obtenido de http://www.kaggle.com

Titanic Lab

Ejercicio de predicción

A partir de un *training set* que contiene los datos de la población de personas que abordaron el Titanic en 31 de Mayo de 1911, se intenta predecir cual habria sido la probabilidad de salvarse en el hundimiento si alguien tuviera un género específico, una litera en una clase específica, con una edad específica, etc.

Este ejercicio hace parte de los retos públicos del sitio http://www.kaggle.com y cada estudiante deberá realizar una predicción y depositar sus resultados en el formulario de esa página web para que reciba una ponderación.

Objetivo

Ademas de ilustrar como utilizar *R* para procesar un conjunto de datos, este documento es un ejemplo de la manera como esperamos que los estudianties del curso de **Analisis y Visualización de Datos** entreguen sus tareas o reportes utilizando *Markdown*. Markdown permite dar formato a documentos planos para exportarlos como HTML, PDF y otros formatos de distribución convencionales usando la librería `knitr` de *R*.

Para tener una referencia de como darle formato a sus archivos en Markdown consultar los ejemplos que hay en Internet. Se recomienda especialmente revisar https://www.rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf.

Pregunta de investigacion

Las mujeres y los niños tuvieron prioridad en los procedimientos de salvamento del Titanic?

Una estimación de base muy burda es simplemente contar cuantas personas sobrevivieron y predecir la superviviencia con respecto a la mayoría. (Ver predicción base pag. 9).

Si hacemos un conteo del sexo y la edad de las víctimas podemos tener otra estimación de cuantas mujeres y niños se salvaron, pero esa observación no se puede generalizar sin error porque es probable que una mujer que abordó en tercera clase tuviera menos chance de sobrevivir que un hombre de primera clase.

Una respuesta prematura observando conteos simples puede ser **si** o **no**. Lo interesante de esta pregunta es saber que circunstancias aumentan la prioridad de las mujeres y los niños.

Procedimiento de lectura de datos desde un archivo .csv

Asumiendo que los archivos están en la carpeta del proyecto en su *working directory*, se revisan los valores faltantes y las dimensiones de los dataframes

```
# El *working directory* se puede obtener con el metodo *getwd()* y se puede
# asignar con *setwd(URL)* Leemos los archivos y los asignamos a dos
# dataframes
train <- read.csv("train.csv")
test <- read.csv("test.csv")
```

Los tipos de datos para todos los campos (columna) de un dataset puede revisarse cuando se pide su estructura usando `str(dataframe)`.

```
# Revise los contenidos de cada dataset
str(train)
```

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 354 273 16 555 516 625 413 57
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
```

```
str(test)
```

```
## 'data.frame':    418 obs. of  11 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : Factor w/ 418 levels "Abbott, Master. Eugene Joseph",..: 207 404 270 409 179 367 85 5
##  $ 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 ...
##  $ Ticket     : Factor w/ 363 levels "110469","110489",..: 153 222 74 148 139 262 159 85 101 270 ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : Factor w/ 77 levels "","A11","A18",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1 3 ...
```

En los datos anteriores se ve que los campos de los datasets tienen tipos de datos numérico (num), entero (int) y categórico (Factor). También se pueden pedir el tipo de dato de cada campo así:

```
# Tipo de datos de las columnas
class(train$Sex)
```

```
## [1] "factor"
```

La descripción general de la distribución de los datos de un campo se obtiene invocando `summary(df)`. Aqui se revisa si hay valores atípicos

```
# Revise si hay valores at<U+00ED>picos. En este caso se puede revisar que
# hay *missing values*, *NA's* o cadenas vacias
summary(train)
```

```
##   PassengerId       Survived          Pclass
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000
##  Median :446.0   Median :0.0000   Median :3.000
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##                                     Name         Sex           Age
##  Abbing, Mr. Anthony                   :  1   female:314   Min.   : 0.42
##  Abbott, Mr. Rossmore Edward           :  1   male  :577   1st Qu.:20.12
##  Abbott, Mrs. Stanton (Rosa Hunt)      :  1                Median :28.00
##  Abelson, Mr. Samuel                   :  1                Mean   :29.70
##  Abelson, Mrs. Samuel (Hannah Wizosky):  1                3rd Qu.:38.00
##  Adahl, Mr. Mauritz Nils Martin        :  1                Max.   :80.00
##  (Other)                               :885                NA's   :177
##      SibSp           Parch             Ticket         Fare
##  Min.   :0.000   Min.   :0.0000   1601    :  7   Min.   :  0.00
##  1st Qu.:0.000   1st Qu.:0.0000   347082  :  7   1st Qu.:  7.91
##  Median :0.000   Median :0.0000   CA. 2343:  7   Median : 14.45
##  Mean   :0.523   Mean   :0.3816   3101295 :  6   Mean   : 32.20
##  3rd Qu.:1.000   3rd Qu.:0.0000   347088  :  6   3rd Qu.: 31.00
##  Max.   :8.000   Max.   :6.0000   CA 2144 :  6   Max.   :512.33
##                                   (Other) :852
##          Cabin     Embarked
##             :687    :  2
##  B96 B98    :  4   C:168
##  C23 C25 C27:  4   Q: 77
##  G6         :  4   S:644
##  C22 C26    :  3
##  D          :  3
##  (Other)    :186
```

```
summary(test)
```

```
##   PassengerId         Pclass
##  Min.   : 892.0   Min.   :1.000
```

```
##  1st Qu.: 996.2   1st Qu.:1.000
##  Median :1100.5   Median :3.000
##  Mean   :1100.5   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.:3.000
##  Max.   :1309.0   Max.   :3.000
##
##                                          Name         Sex
##  Abbott, Master. Eugene Joseph            :  1   female:152
##  Abelseth, Miss. Karen Marie              :  1   male  :266
##  Abelseth, Mr. Olaus Jorgensen            :  1
##  Abrahamsson, Mr. Abraham August Johannes :  1
##  Abrahim, Mrs. Joseph (Sophie Halaut Easu):  1
##  Aks, Master. Philip Frank                :  1
##  (Other)                                  :412
##       Age            SibSp           Parch            Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.0000   PC 17608:  5
##  1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.0000   113503  :  4
##  Median :27.00   Median :0.0000   Median :0.0000   CA. 2343:  4
##  Mean   :30.27   Mean   :0.4474   Mean   :0.3923   16966   :  3
##  3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.0000   220845  :  3
##  Max.   :76.00   Max.   :8.0000   Max.   :9.0000   347077  :  3
##  NA's   :86                                        (Other) :396
##       Fare                    Cabin     Embarked
##  Min.   :  0.000                 :327   C:102
##  1st Qu.:  7.896   B57 B59 B63 B66:  3   Q: 46
##  Median : 14.454   A34           :  2   S:270
##  Mean   : 35.627   B45           :  2
##  3rd Qu.: 31.500   C101          :  2
##  Max.   :512.329   C116          :  2
##  NA's   :1         (Other)       : 80
```

```
# Cantidad de filas y columnas
dim(train)
```

```
## [1] 891  12
```

```
dim(test)
```

```
## [1] 418  11
```

Revision de algunos datos del **test** dataset. Falta un registro del listado de tarifas pagadas

```
which(is.na(test$Fare))
```

```
## [1] 153
```

```
test[is.na(test$Fare), ]
```

```
##     PassengerId Pclass               Name  Sex  Age SibSp Parch Ticket
## 153        1044      3 Storey, Mr. Thomas male 60.5     0     0   3701
##     Fare Cabin Embarked
## 153   NA              S
```

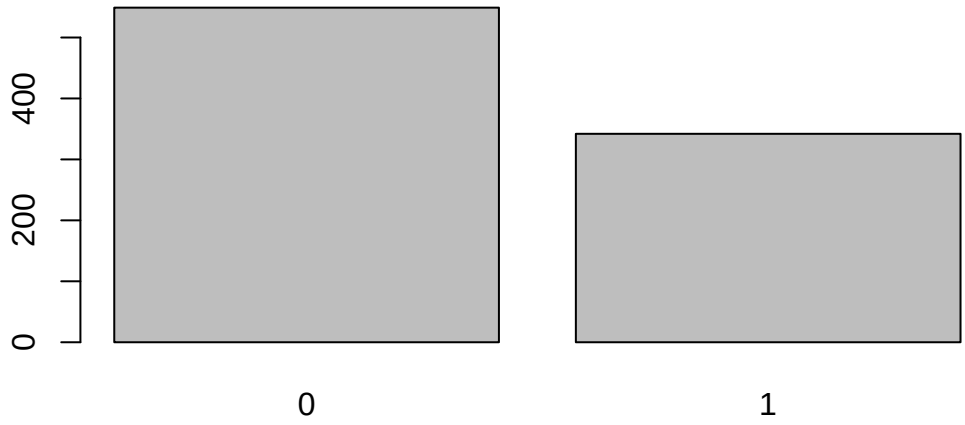
PREDICCIÓN BASE

Revisamos los sobrevivientes:

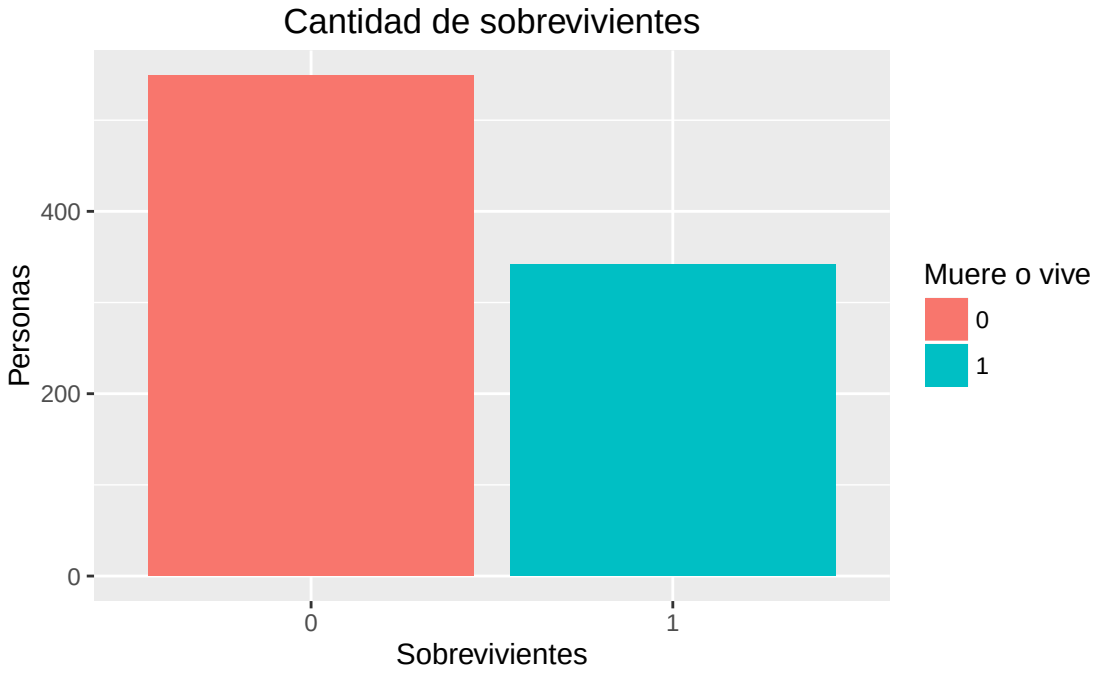
```
# 0 muri<U+00F3>, 1 vivi<U+00F3>
table(train$Survived)
```

```
##
##   0   1
## 549 342
```

```
# Histogramas Usando el graficador nativo de R el resultado es:
plot(as.factor(train$Survived))
```




```
# Pero de ahora en adelante usaremos la libreria ggplot2
ggplot(train, aes(as.factor(Survived), fill = as.factor(Survived))) + geom_bar() +
    labs(title = "Cantidad de sobrevivientes", x = "Sobrevivientes", y = "Personas",
        fill = "Muere o vive")
```



```r
# Los mismos datos como proporciones
prop.table(table(train$Survived))
```

```
##
##         0         1
## 0.6161616 0.3838384
```

Predicción que todos mueren. Esta es la linea de base de predicción (La peor predicción con respecto al training set)

```r
# Se crea una columna predicci<U+00F3>n
train$Prediction <- 0
# Confusion matrix. Recibe lo predicho y lo compara con los datos reales.
confusionMatrix(train$Prediction, train$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 549 342
##          1   0   0
##
##                Accuracy : 0.6162
##                  95% CI : (0.5833, 0.6482)
##     No Information Rate : 0.6162
##     P-Value [Acc > NIR] : 0.5148
##
##                   Kappa : 0
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 1.0000
##             Specificity : 0.0000
##          Pos Pred Value : 0.6162
##          Neg Pred Value :    NaN
##              Prevalence : 0.6162
##          Detection Rate : 0.6162
##    Detection Prevalence : 1.0000
##       Balanced Accuracy : 0.5000
##
##        'Positive' Class : 0
##
```

Vemos que la precisión es del 61.62% (0.6162)

Para enviar este archivo a http://www.kaggle.com se debería hacer lo siguiente en el dataset *test*

```r
# Creacion de columna survived y la lleno con 0 para los 418 filas
test$Survived <- rep(0, 418)
resultados <- data.frame(PassengerId = test$PassengerId, Survived = test$Survived)
# Creo el archivo con los resultados.
```

```r
write.csv(resultados, file = "todosMueren.csv", row.names = FALSE)
# El archivo 'todosMueren.csv' se envia a Kaggle
```

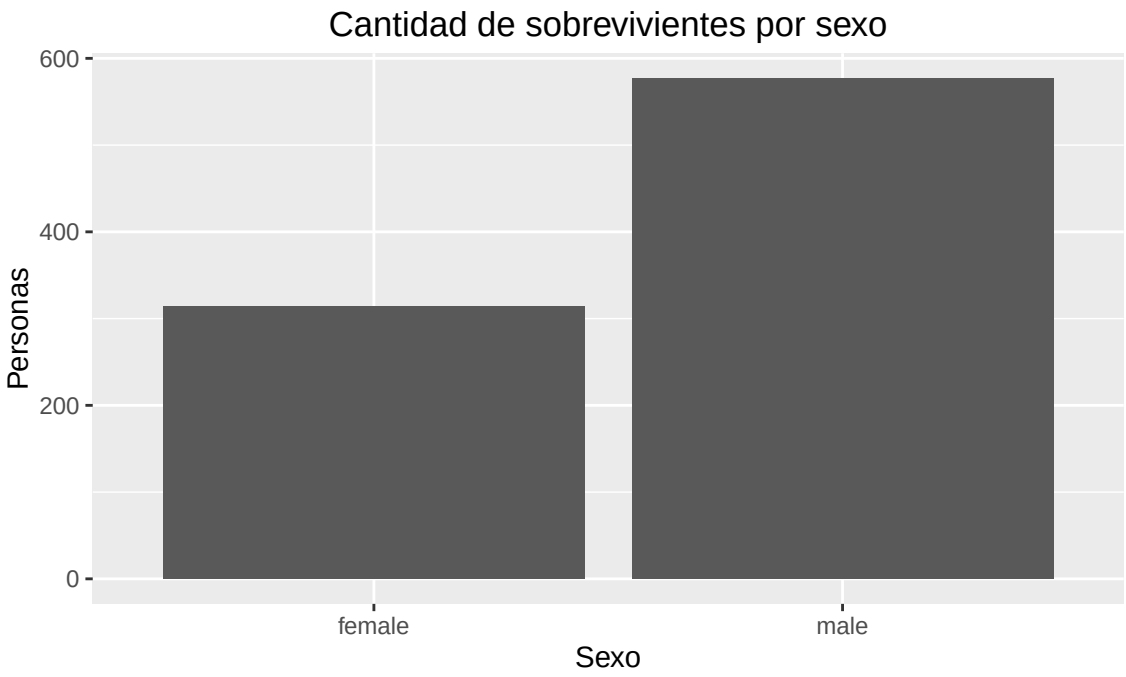
MUJERES Y NIÑOS PRIMERO !

A continuación exploramos los datos que nos permitan acercanos a una respuesta a la pregunta de investigación. Obtenemos una table de las personas que sobrevivieron que están en el **training set**.

Mujeres

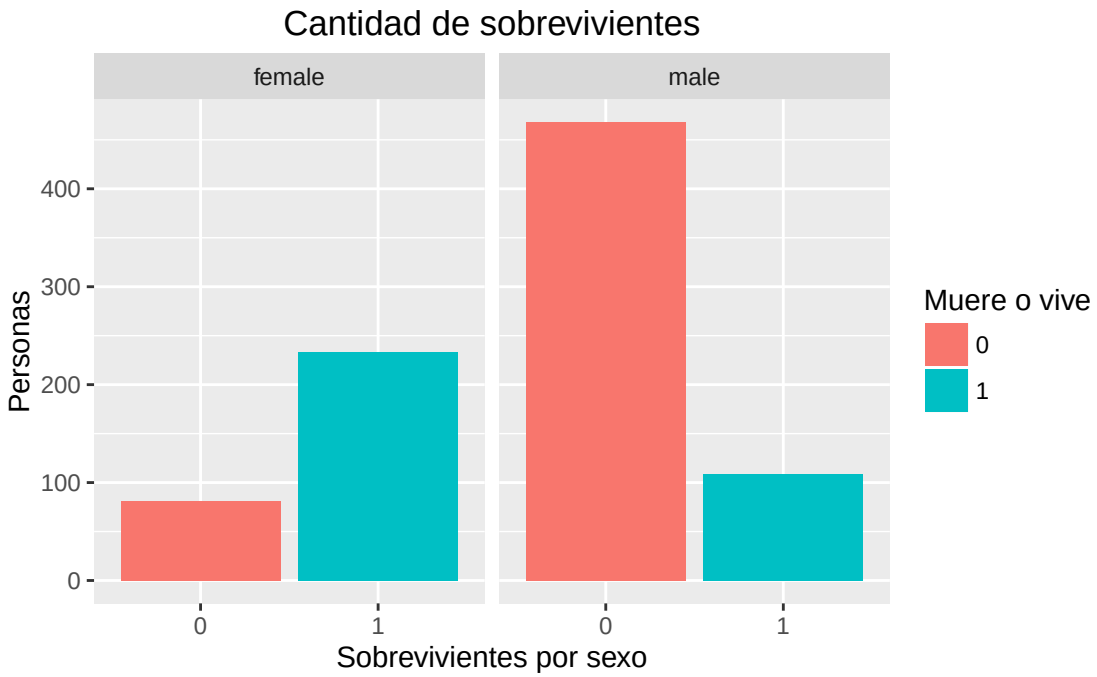
Miramos la ditribución de hombres y mujeres que abordaron el barco

```r
ggplot(train, aes(Sex, fill = Survived)) + geom_bar() + labs(title = "Cantidad de sobrevivientes por se
    x = "Sexo", y = "Personas")
```



Sobrevivientes y sexo:

```r
ggplot(train, aes(as.factor(Survived), fill = as.factor(Survived))) + geom_bar() +
    facet_grid(. ~ Sex) + labs(title = "Cantidad de sobrevivientes", x = "Sobrevivientes por sexo",
    y = "Personas", fill = "Muere o vive")
```




```
summary(train$Sex)
```

```
## female   male
##    314    577
```

```
# Muestra la proporcion con respecto al total de los datos
prop.table(table(train$Sex, train$Survived))
```

```
##
##                    0          1
##   female 0.09090909 0.26150393
##   male   0.52525253 0.12233446
```

```
# Muestra la proporcion agregada por filas
prop.table(table(train$Sex, train$Survived), 1)
```

```
##
##                    0         1
##   female 0.2579618 0.7420382
##   male   0.8110919 0.1889081
```

```
counts <- table(train$Survived, train$Sex)

# Calcula las proporciones de supervivencia a mano. Obtenemos los mismos
# datos que en el paso anterior.
counts[2]/(counts[1] + counts[2])
```

```
## [1] 0.7420382
```

```r
counts[4]/(counts[3] + counts[4])
```

```
## [1] 0.1889081
```

```r
# Hacemos un una nueva prediccion: Todos mueren:
train$Prediction <- 0
# Pero las muejres viven:
train$Prediction[train$Sex == "female"] <- 1
confusionMatrix(train$Prediction, train$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 468 109
##          1  81 233
##
##                Accuracy : 0.7868
##                  95% CI : (0.7584, 0.8132)
##     No Information Rate : 0.6162
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.5421
##  Mcnemar's Test P-Value : 0.05014
##
##             Sensitivity : 0.8525
##             Specificity : 0.6813
##          Pos Pred Value : 0.8111
##          Neg Pred Value : 0.7420
##              Prevalence : 0.6162
##          Detection Rate : 0.5253
##    Detection Prevalence : 0.6476
##       Balanced Accuracy : 0.7669
##
##        'Positive' Class : 0
##
```

Simplemente con predecir que las mujeres sobreviven la precisión aumenta a 78.68%.

```r
# Generamos el archivo para Kaggle
test$Survived <- 0
test$Survived[test$Sex == "female"] <- 1
resultados <- data.frame(PassengerId = test$PassengerId, Survived = test$Survived)
write.csv(resultados, file = "mujeres.csv", row.names = FALSE)
```

Niños

Revision de distribución por edades

```
# Miramos qui<U+00E9>nes y cu<U+00E1>ntos pasajeros no tienen la edad
# registrada, muestra los primeros 6 registros del dataframe
head(train[is.na(train$Age), ])
```

```
##    PassengerId Survived Pclass                          Name    Sex Age
## 6            6        0      3              Moran, Mr. James   male  NA
## 18          18        1      2  Williams, Mr. Charles Eugene   male  NA
## 20          20        1      3       Masselmani, Mrs. Fatima female  NA
## 27          27        0      3       Emir, Mr. Farred Chehab   male  NA
## 29          29        1      3 O'Dwyer, Miss. Ellen "Nellie" female  NA
## 30          30        0      3           Todoroff, Mr. Lalio   male  NA
##    SibSp Parch Ticket    Fare Cabin Embarked Prediction
## 6      0     0 330877  8.4583              Q          0
## 18     0     0 244373 13.0000              S          0
## 20     0     0   2649  7.2250              C          1
## 27     0     0   2631  7.2250              C          0
## 29     0     0 330959  7.8792              Q          1
## 30     0     0 349216  7.8958              S          0
```

```
# En total los pasajeros sin edad son...
nrow(train[is.na(train$Age), ])
```

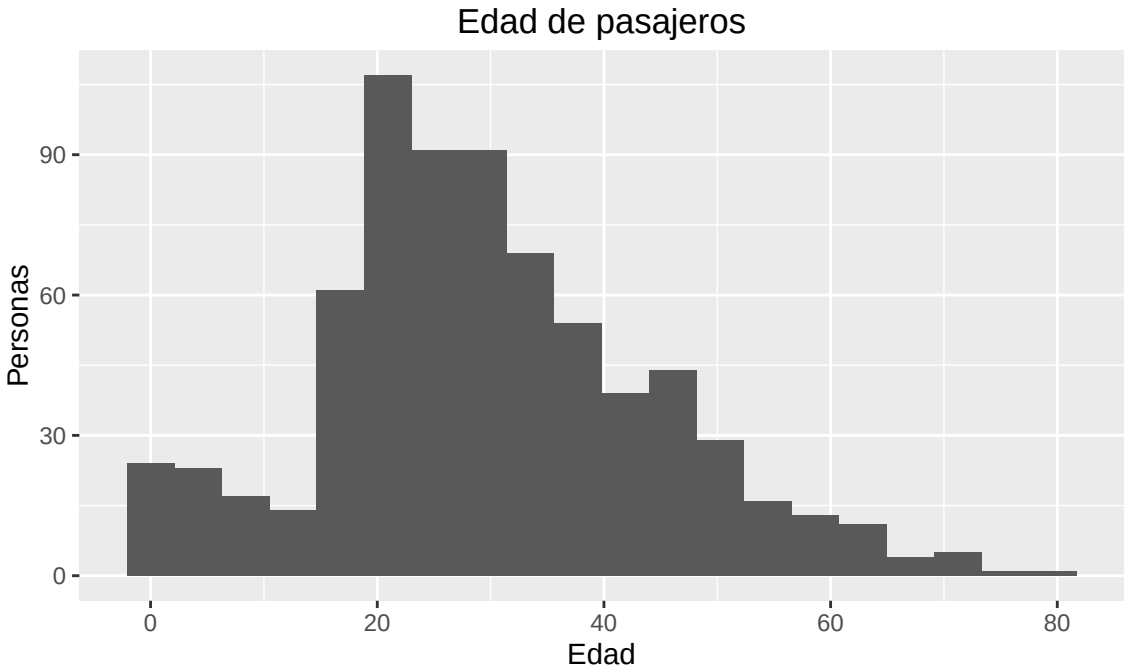
```
## [1] 177
```

```
# y estan en estas posiones del dataframe:
which(is.na(train$Age))
```

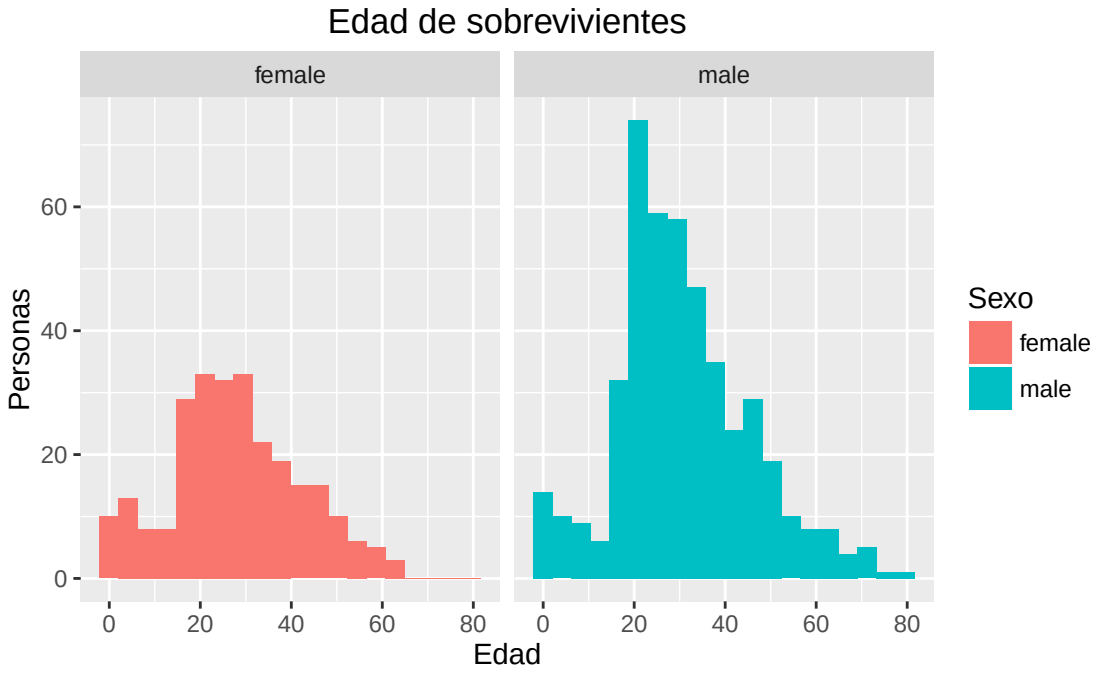
```
##   [1]   6  18  20  27  29  30  32  33  37  43  46  47  48  49  56  65  66
##  [18]  77  78  83  88  96 102 108 110 122 127 129 141 155 159 160 167 169
##  [35] 177 181 182 186 187 197 199 202 215 224 230 236 241 242 251 257 261
##  [52] 265 271 275 278 285 296 299 301 302 304 305 307 325 331 335 336 348
##  [69] 352 355 359 360 365 368 369 376 385 389 410 411 412 414 416 421 426
##  [86] 429 432 445 452 455 458 460 465 467 469 471 476 482 486 491 496 498
## [103] 503 508 512 518 523 525 528 532 534 539 548 553 558 561 564 565 569
## [120] 574 579 585 590 594 597 599 602 603 612 613 614 630 634 640 644 649
## [137] 651 654 657 668 670 675 681 693 698 710 712 719 728 733 739 740 741
## [154] 761 767 769 774 777 779 784 791 793 794 816 826 827 829 833 838 840
## [171] 847 850 860 864 869 879 889
```

Un histograma de las edades muestra que la poblacion mas grande estaba cerca a los 20 años de edad. Pero hay diferencias importantes en el sexo.

```
ggplot(train, aes(Age)) + geom_histogram(bins = 20) + labs(title = "Edad de pasajeros",
    x = "Edad", y = "Personas")
```




```
# Diferencia de edades por sexo
ggplot(train, aes(Age, fill = Sex)) + geom_histogram(bins = 20) + facet_grid(. ~
    Sex) + labs(title = "Edad de sobrevivientes", x = "Edad", y = "Personas",
    fill = "Sexo")
```



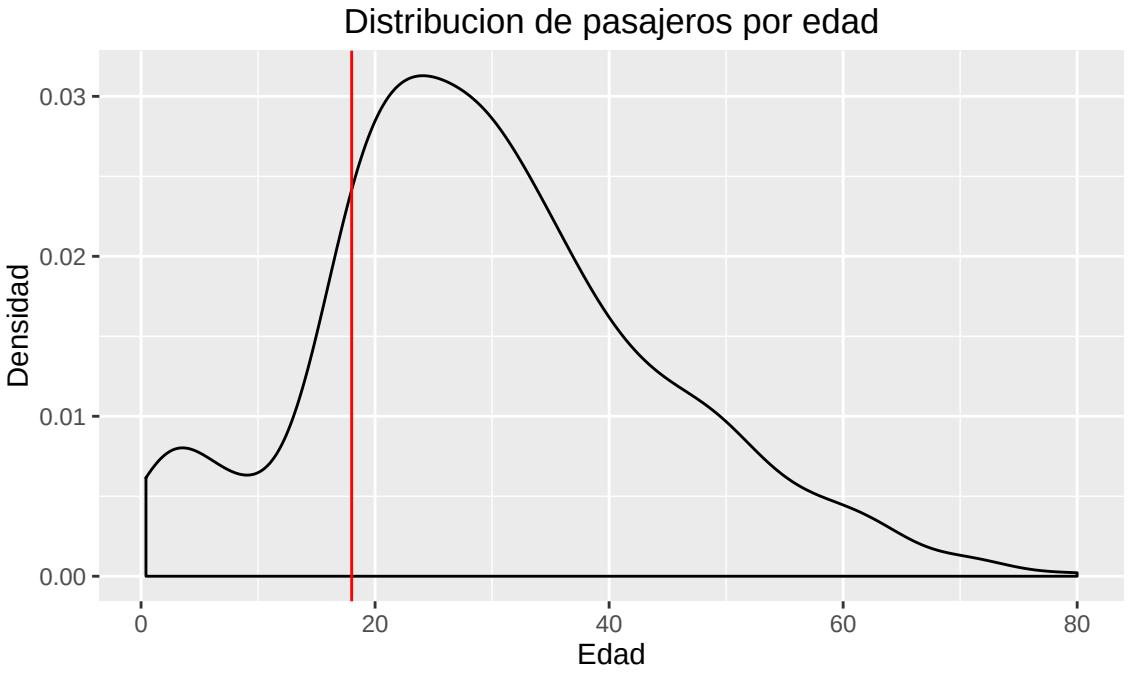
Arriba veíamos la distribución de la edad que describe una curva relativamente normal.

```
summary(train$Age)  #177 missing values
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
```

```
##    0.42   20.12   28.00   29.70   38.00   80.00     177
```

```r
# Aqui vemos claramente que la distribuci<U+00F3>n de edad es bimodal. La
# linea roja indica edad = 18 a<U+00F1>os
ggplot(train, aes(Age)) + geom_density() + geom_vline(xintercept = 18, color = "red") +
    labs(title = "Distribucion de pasajeros por edad", x = "Edad", y = "Densidad")
```




```r
# Convertir variable num<U+00E9>rica a binaria Asignamos 0 para decir que
# todos los pasejeros no son ni<U+00F1>os. Incluso los *Missing Values* de
# edad
train$Child <- 0
# Todo pasajero < 18 es ni<U+00F1>o
train$Child[train$Age < 18] <- 1
# Matriz de confusion
train$Prediction <- 0
# todo ni<U+00F1>o vive
train$Prediction[train$Child == 1] <- 1
confusionMatrix(train$Prediction, train$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 497 281
##          1  52  61
##
##                Accuracy : 0.6263
##                  95% CI : (0.5936, 0.6581)
##     No Information Rate : 0.6162
##     P-Value [Acc > NIR] : 0.2797
##
```

```
##                   Kappa : 0.0957
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9053
##             Specificity : 0.1784
##          Pos Pred Value : 0.6388
##          Neg Pred Value : 0.5398
##              Prevalence : 0.6162
##          Detection Rate : 0.5578
##    Detection Prevalence : 0.8732
##       Balanced Accuracy : 0.5418
##
##        'Positive' Class : 0
##
```

Insight: Discriminar solo por edad no aporta mucho porque la precisión de 62.63 % es ligeramente superior a la línea base.

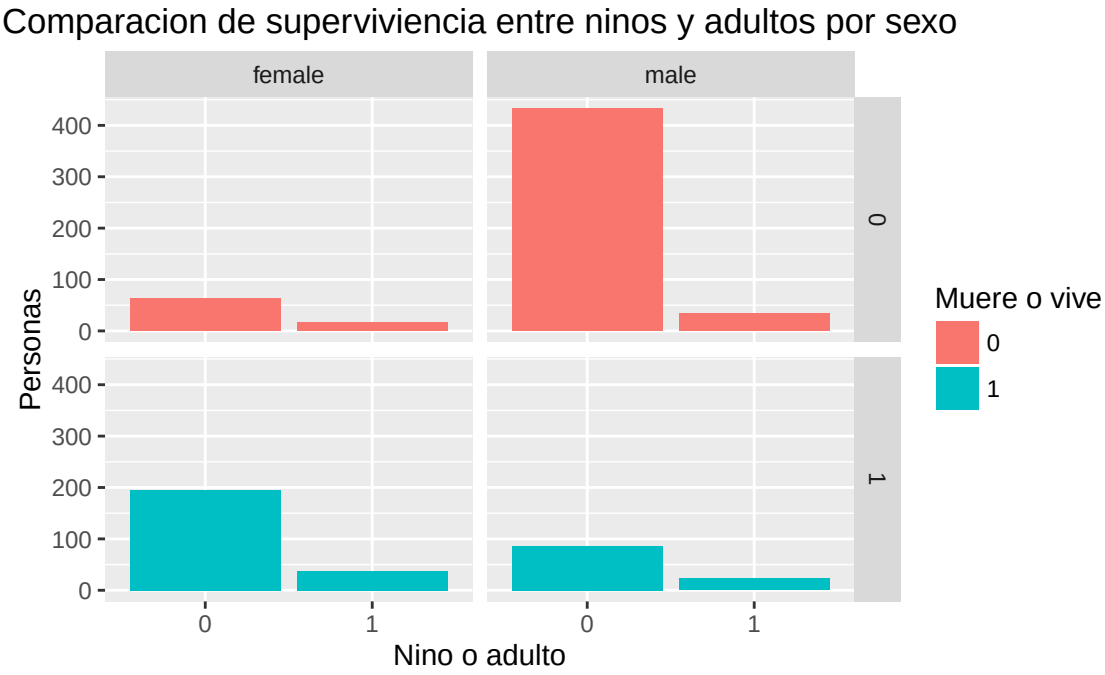
Distribuciones por sexo y edad

Revisamos la cantidad de mujeres y hombres sobrevivientes, y cuantos de ellos son niñas o niños.

```
# Cu<U+00E1>ntos sobreviven?
aggregate(Survived ~ Child + Sex, data = train, FUN = sum)
```

```
##   Child    Sex Survived
## 1     0 female      195
## 2     1 female       38
## 3     0   male       86
## 4     1   male       23
```

```
# En gr<U+00E1>fica:
ggplot(train, aes(as.factor(Child), fill = as.factor(Survived))) + geom_bar() +
    facet_grid(Survived ~ Sex) + labs(title = "Comparacion de superviviencia entre ninos y adultos por s
    x = "Nino o adulto", y = "Personas", fill = "Muere o vive")
```




```
# Qu<U+00E9> proporciones?
aggregate(Survived ~ Child + Sex, data = train, FUN = function(x) {
    sum(x)/length(x)
})
```

```
##   Child    Sex  Survived
## 1     0 female 0.7528958
## 2     1 female 0.6909091
## 3     0   male 0.1657033
## 4     1   male 0.3965517
```

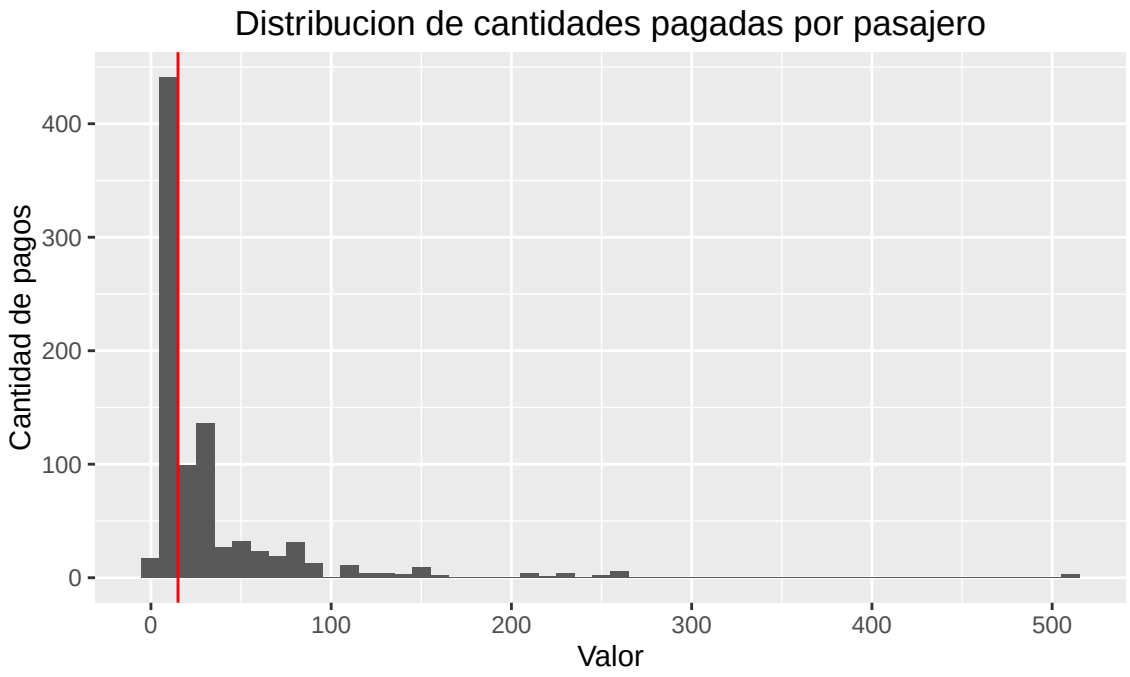
Vemos que los hombres así sean niños siempre van a tener mas probabilidad de morir y las mujeres niñas o adultas de vivir.

PRECIO Y CLASE DEL PASAJE

```
# PRECION DEL BILLETE
summary(train$Fare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    7.91   14.45   32.20   31.00  512.30
```

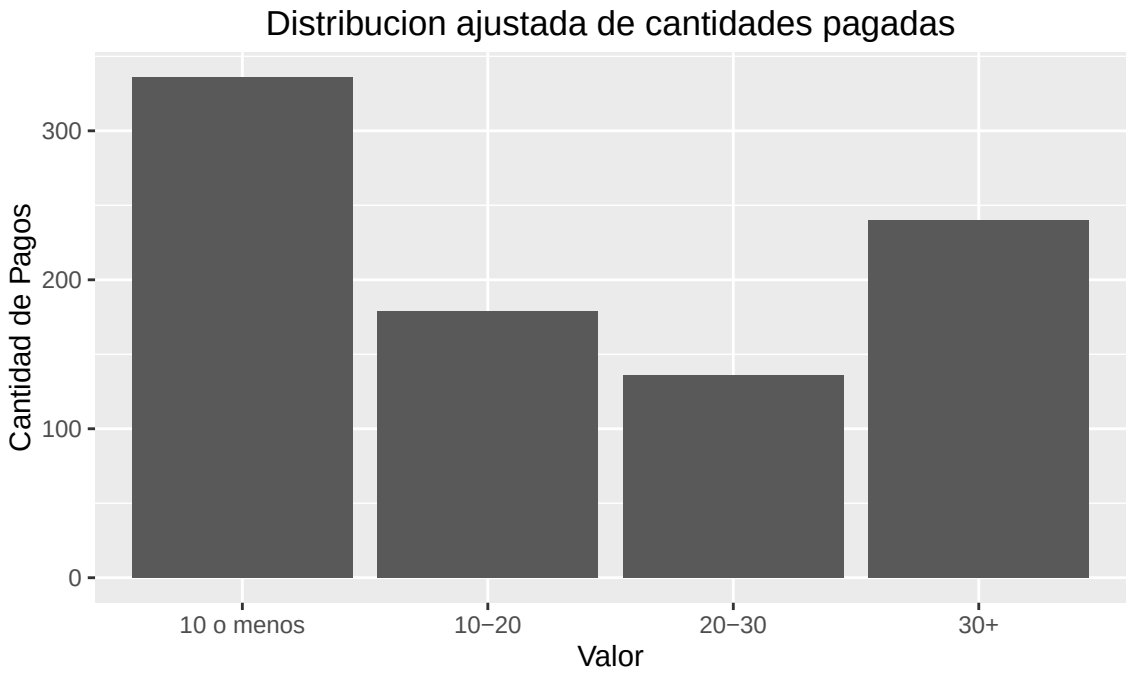
```
# table(train$Fare)
ggplot(train, aes(Fare)) + geom_histogram(binwidth = 10) + geom_vline(xintercept = 15,
    color = "red") + labs(title = "Distribucion de cantidades pagadas por pasajero",
    x = "Valor", y = "Cantidad de pagos")
```



Lo que podemos ver es que la mayoría de los pasajeros pagaron menos de $15 dólares. Hay que tener en cuenta que esta variable representa el total pagado, es decir que una persona que pago un pasaje tiene un valor pagado acumulado menor al de una persona que pagó todos los pasajes de su familia.

Vemos que hay demasiados valores posible por lo tanto decidimos discretizar la variable en 4 rangos: menos de $10, entre $10 y $20, entre $20 y $30, mas de $30

```
# Se crean bins para rangos de precios. Se deterimnan los rangos para
# distribuir la mayor cantidad de valores bajo la curva
train$Fare2 <- "30+"
train$Fare2[train$Fare < 30 & train$Fare >= 20] <- "20-30"
train$Fare2[train$Fare < 20 & train$Fare >= 10] <- "10-20"
train$Fare2[train$Fare < 10] <- "10 o menos"
ggplot(train, aes(Fare2)) + geom_bar() + labs(title = "Distribucion ajustada de cantidades pagadas",
    x = "Valor", y = "Cantidad de Pagos", fill = "Muere o vive")
```

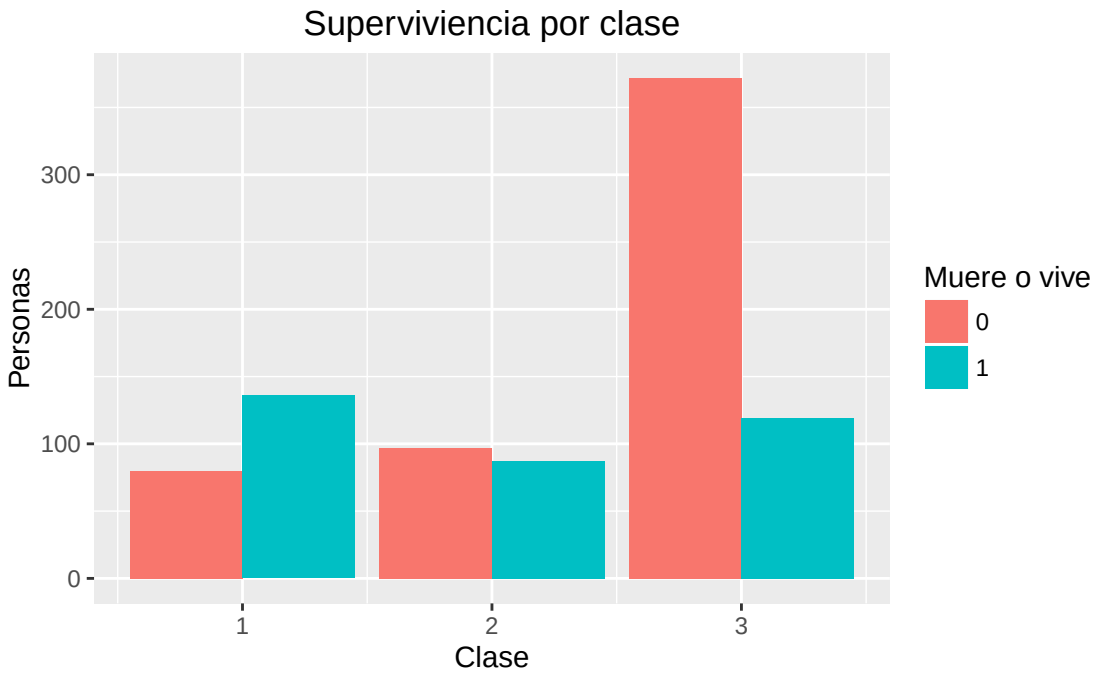



```
table(train$Fare2)
```

```
##
## 10 o menos      10-20      20-30        30+
##        336        179        136        240
```

Revisamos la supervivencia por clase

```
# Clase
Pclass_survival <- table(train$Survived, train$Pclass)
ggplot(train, aes(Pclass)) + geom_bar(aes(fill = as.factor(Survived)), position = "dodge") +
    labs(title = "Superviviencia por clase", x = "Clase", y = "Personas", fill = "Muere o vive")
```




```
Pclass_survival[2]/(Pclass_survival[1] + Pclass_survival[2])
```

```
## [1] 0.6296296
```

```
Pclass_survival[4]/(Pclass_survival[3] + Pclass_survival[4])
```

```
## [1] 0.4728261
```

```
Pclass_survival[6]/(Pclass_survival[5] + Pclass_survival[6])
```

```
## [1] 0.2423625
```

Vemos que hay mas probabilidad de sobrevivir en clase 1 (62.96%) que en clase 2 (47.28%) y que en clase 3 (24.23%). Por lo tanto incluimos la clase y la tarifa en nuestro análisis.

Revisamos la distribución por tarifa, clase, sexo y supervivencia para determinar que algunas mujeres tienen mas probabilidad de morir.

```
aggregate(Survived ~ Fare2 + Pclass + Sex, data = train, FUN = function(x) {
    sum(x)/length(x)
})
```

```
##         Fare2 Pclass    Sex  Survived
## 1       20-30      1 female 0.8333333
## 2         30+      1 female 0.9772727
## 3       10-20      2 female 0.9142857
## 4       20-30      2 female 0.9000000
## 5         30+      2 female 1.0000000
## 6  10 o menos      3 female 0.5937500
## 7       10-20      3 female 0.5813953
```

```
## 8       20-30      3 female 0.3333333
## 9         30+      3 female 0.1250000
## 10 10 o menos      1   male 0.0000000
## 11      20-30      1   male 0.4000000
## 12        30+      1   male 0.3837209
## 13 10 o menos      2   male 0.0000000
## 14      10-20      2   male 0.1587302
## 15      20-30      2   male 0.1600000
## 16        30+      2   male 0.2142857
## 17 10 o menos      3   male 0.1115385
## 18      10-20      3   male 0.2368421
## 19      20-30      3   male 0.1250000
## 20        30+      3   male 0.2400000
```

```
# Ajusto mi prediccion con base en lo anterior
train$Prediction <- 0
train$Prediction[train$Sex == "female"] <- 1
# Algunas mujeres mueren: Las que viajan en tercera clase y pagaron $20 o
# mas por su pasaje
train$Prediction[train$Sex == "female" & train$Pclass == 3 & train$Fare >= 20] <- 0
confusionMatrix(train$Prediction, train$Survived)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 496 118
##          1  53 224
##
##                Accuracy : 0.8081
##                  95% CI : (0.7807, 0.8334)
##     No Information Rate : 0.6162
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5792
##  Mcnemar's Test P-Value : 9.871e-07
##
##             Sensitivity : 0.9035
##             Specificity : 0.6550
##          Pos Pred Value : 0.8078
##          Neg Pred Value : 0.8087
##              Prevalence : 0.6162
##          Detection Rate : 0.5567
##    Detection Prevalence : 0.6891
##       Balanced Accuracy : 0.7792
##
##        'Positive' Class : 0
##
```

Ajustar la predicción teniendo en cuenta la clase y el precio del pasaje implica una mejora en la predicción logrando una precisión del 80.81%. Hay que tener en cuenta que estos resultados son con respecto al training set, lo que quealmente importa es la robustez del modelo con respecto al test set. Eventualmente podriamos *sobreaprender* (overfitting) si nuestras reglas de particionamiento se vuelven demasiado específicas (no generalizables).

```r
# Preparar el archivo para enviar a www.kaggle.com
test$Survived <- 0
test$Survived[test$Sex == "female"] <- 1
test$Survived[test$Sex == "female" & test$Pclass == 3 & test$Fare >= 20] <- 0
resultados <- data.frame(PassengerId = test$PassengerId, Survived = test$Survived)
write.csv(resultados, file = "mujeresDeClase.csv", row.names = FALSE)
```

TAREA:

Qué pasaría si se incluye la edad y/o el número de familiares que viajan con cada pasajero?

Para la próxima clase deben traer: 1) Una breve explicación en Markdown de lo que hicieron.
2) Las matrices de confusión de sus predicciones utilizando el training set. 3) Los resultados que les arroja kaggle cuando envien los archivos test set de sus predecciones (captura de pantalla).