Minería de datos: PEC1

Autor: Nombre estudiante

Marzo 2020

Contents

Introducción

Presentación

Esta prueba de evaluación continuada cubre el módulo 1,2 y 8 del programa de la asignatura.

Competencias

Las competencias que se trabajan en esta prueba son:

- Uso y aplicación de las TIC en el ámbito académico y profesional
- Capacidad para innovar y generar nuevas ideas.
- Capacidad para evaluar soluciones tecnológicas y elaborar propuestas de proyectos teniendo en cuenta los recursos, las alternativas disponibles y las condiciones de mercado.

- Conocer las tecnologías de comunicaciones actuales y emergentes, así como saberlas aplicar convenientemente para diseñar y desarrollar soluciones basadas en sistemas y tecnologías de la información.
- Aplicación de las técnicas específicas de ingeniería del software en las diferentes etapas del ciclo de vida de un proyecto.
- Capacidad para aplicar las técnicas específicas de tratamiento, almacenamiento y administración de datos.
- Capacidad para proponer y evaluar diferentes alternativas tecnológicas para resolver un problema concreto.
- Capacidad de utilizar un lenguaje de programación.
- Capacidad para desarrollar en una herramienta IDE.
- Capacidad de plantear un proyecto de minería de datos.

Objetivos

- Asimilar correctamente el módulo 1 y 2.
- Qué es y qué no es MD.
- Ciclo de vida de los proyectos de MD.
- Diferentes tipologías de MD.
- Conocer las técnicas propias de una fase de preparación de datos y objetivos a alcanzar.

Descripción de la PEC a realizar

La prueba está estructurada en 1 ejercicio teórico/práctico y 1 ejercicio práctico que pide que se desarrolle la fase de preparación en un juego de datos.
Deben responderse todos los ejercicios para poder superar la PEC.

Recursos

Para realizar esta práctica recomendamos la lectura de los siguientes documentos:

- Módulo 1, 2 y 8 del material didáctico.
- RStudio Cheat Sheet: Disponible en el aula Laboratorio de Minería de datos.
- R Base Cheat Sheet: Disponible en el aula Laboratorio de Minería de datos.

Criterios de evaluación

Ejercicios teóricos
Todos los ejercicios deben ser presentados de forma razonada y clara, especificando todos y cada uno de los pasos que se hayan llevado a cabo para su resolución. No se aceptará ninguna respuesta que no esté claramente justificada.

Ejercicios prácticos
Para todas las PEC es necesario documentar en cada apartado del ejercicio práctico qué se ha hecho y cómo se ha hecho.

Formato y fecha de entrega

El formato de entrega es: usernameestudiant-PECn.html y rmd
Fecha de Entrega: 01/04/2020
Se debe entregar la PEC en el buzón de entregas del aula

Nota: Propiedad intelectual

A menudo es inevitable, al producir una obra multimedia, hacer uso de recursos creados por terceras personas. Es por lo tanto comprensible hacerlo en el marco de una práctica de los estudios de Informática, Multimedia y Telecomunicación de la UOC, siempre y cuando esto se documente claramente y no suponga plagio en la práctica.

Por lo tanto, al presentar una práctica que haga uso de recursos ajenos, se debe presentar junto con ella un documento en qué se detallen todos ellos, especificando el nombre de cada recurso, su autor, el lugar dónde se obtuvo y su estatus legal: si la obra está protegida por el copyright o se acoge a alguna otra licencia de uso (Creative Commons, licencia GNU, GPL ...). El estudiante deberá asegurarse de que la licencia no impide específicamente su uso en el marco de la práctica. En caso de no encontrar la información correspondiente tendrá que asumir que la obra está protegida por copyright.

Deberéis, además, adjuntar los ficheros originales cuando las obras utilizadas sean digitales, y su código fuente si corresponde.

Enunciado

Como ejemplo, trabajaremos con el conjunto de datos "Titanic" que recoge datos sobre el famoso crucero y sobre el que es fácil realizar tareas de clasificación predictiva sobre la variable "Survived".

De momento dejaremos para las siguientes prácticas el estudio de algoritmos predictivos y nos centraremos por ahora en el estudio de las variables de una muestra de datos, es decir, haremos un trabajo descriptivo del mismo.

Las actividades que llevaremos a cabo en esta práctica suelen enmarcarse en las fases iniciales de un proyecto de minería de datos y consisten en la selección de características o variables y la preparación del los datos para posteriormente ser consumido por un algoritmo.

Las técnicas que trabajaremos son las siguientes:

1. Normalización
2. Discretización
3. Gestión de valores nulos
4. Estudio de correlaciones
5. Reducción de la dimensionalidad
6. Análisis visual del conjunto de datos

Ejemplo de estudio visual con el juego de datos Titanic

Procesos de limpieza del conjunto de datos

Primer contacto con el conjunto de datos, visualizamos su estructura.

```
# Cargamos los paquetes R que vamos a usar
library(ggplot2)
library(dplyr)

# Cargamos el fichero de datos
totalData <- read.csv('titanic.csv',stringsAsFactors = FALSE)
filas=dim(totalData)[1]

# Verificamos la estructura del conjunto de datos
str(totalData)
```

```
## 'data.frame':    2207 obs. of  11 variables:
##  $ name    : chr  "Abbing, Mr. Anthony" "Abbott, Mr. Eugene Joseph" "Abbott, Mr. Rossmore Edward" "Al
##  $ gender  : chr  "male" "male" "male" "female" ...
##  $ age     : num  42 13 16 39 16 25 30 28 27 20 ...
##  $ class   : chr  "3rd" "3rd" "3rd" "3rd" ...
##  $ embarked: chr  "S" "S" "S" "S" ...
##  $ country : chr  "United States" "United States" "United States" "England" ...
##  $ ticketno: int  5547 2673 2673 2673 348125 348122 3381 3381 2699 3101284 ...
##  $ fare    : num  7.11 20.05 20.05 20.05 7.13 ...
##  $ sibsp   : int  0 0 1 1 0 0 1 1 0 0 ...
##  $ parch   : int  0 2 1 1 0 0 0 0 0 0 ...
##  $ survived: chr  "no" "no" "no" "yes" ...
```

Descripción de las variables contenidas en el fichero:

name a string with the name of the passenger.

gender a factor with levels male and female.

age a numeric value with the persons age on the day of the sinking. The age of babies (under 12 months) is given as a fraction of one year (1/month).

class a factor specifying the class for passengers or the type of service aboard for crew members.

embarked a factor with the persons place of of embarkment.

country a factor with the persons home country.

ticketno a numeric value specifying the persons ticket number (NA for crew members).

fare a numeric value with the ticket price (NA for crew members, musicians and employees of the shipyard company).

sibsp an ordered factor specifying the number if siblings/spouses aboard; adopted from Vanderbild data set.

parch an ordered factor specifying the number of parents/children aboard; adopted from Vanderbild data set.

survived a factor with two levels (no and yes) specifying whether the person has survived the sinking.

Mostramos estadísticas bàsicas y después trabajamos los atributos con valores vacíos.

```
#Estadísticas básicas
summary(totalData)
```

```
##      name              gender              age              class
##  Length:2207        Length:2207        Min.   : 0.1667   Length:2207
##  Class :character   Class :character   1st Qu.:22.0000   Class :character
##  Mode  :character   Mode  :character   Median :29.0000   Mode  :character
##                                        Mean   :30.4367
##                                        3rd Qu.:38.0000
```

```
##                                         Max.   :74.0000
##                                         NA's   :2
##    embarked           country              ticketno            fare
##  Length:2207        Length:2207        Min.   :      2   Min.   :  3.030
##  Class :character   Class :character   1st Qu.:  14262   1st Qu.:  7.181
##  Mode  :character   Mode  :character   Median : 111427   Median : 14.090
##                                        Mean   : 284216   Mean   : 33.405
##                                        3rd Qu.: 347077   3rd Qu.: 31.061
##                                        Max.   :3101317   Max.   :512.061
##                                        NA's   :891       NA's   :916
##      sibsp            parch          survived
##  Min.   :0.0000   Min.   :0.0000   Length:2207
##  1st Qu.:0.0000   1st Qu.:0.0000   Class :character
##  Median :0.0000   Median :0.0000   Mode  :character
##  Mean   :0.4996   Mean   :0.3856
##  3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :8.0000   Max.   :9.0000
##  NA's   :900      NA's   :900
```

```
# Estadísticas de valores vacíos
colSums(is.na(totalData))
```

```
##     name   gender      age    class embarked  country ticketno     fare
##        0        0        2        0        0       81      891      916
##    sibsp    parch survived
##      900      900        0
```

```
colSums(totalData=="")
```

```
##     name   gender      age    class embarked  country ticketno     fare
##        0        0       NA        0        0       NA       NA       NA
##    sibsp    parch survived
##       NA       NA        0
```

```
# Tomamos valor "Desconocido" para los valores vacíos de la variable "country"
totalData$Embarked[totalData$country==""]="Desconocido"

# Tomamos la media para valores vacíos de la variable "Age"
totalData$Age[is.na(totalData$age)] <- mean(totalData$age,na.rm=T)
```

Discretizamos cuando tiene sentido y en función de cada variable.

```
# ¿Con qué variables tendría sentido un proceso de discretización?
apply(totalData,2, function(x) length(unique(x)))
```

```
##     name   gender      age    class embarked  country ticketno     fare
##     2202        2       80        7        4       49      925      277
##    sibsp    parch survived Embarked      Age
##        8        9        2        1        2
```

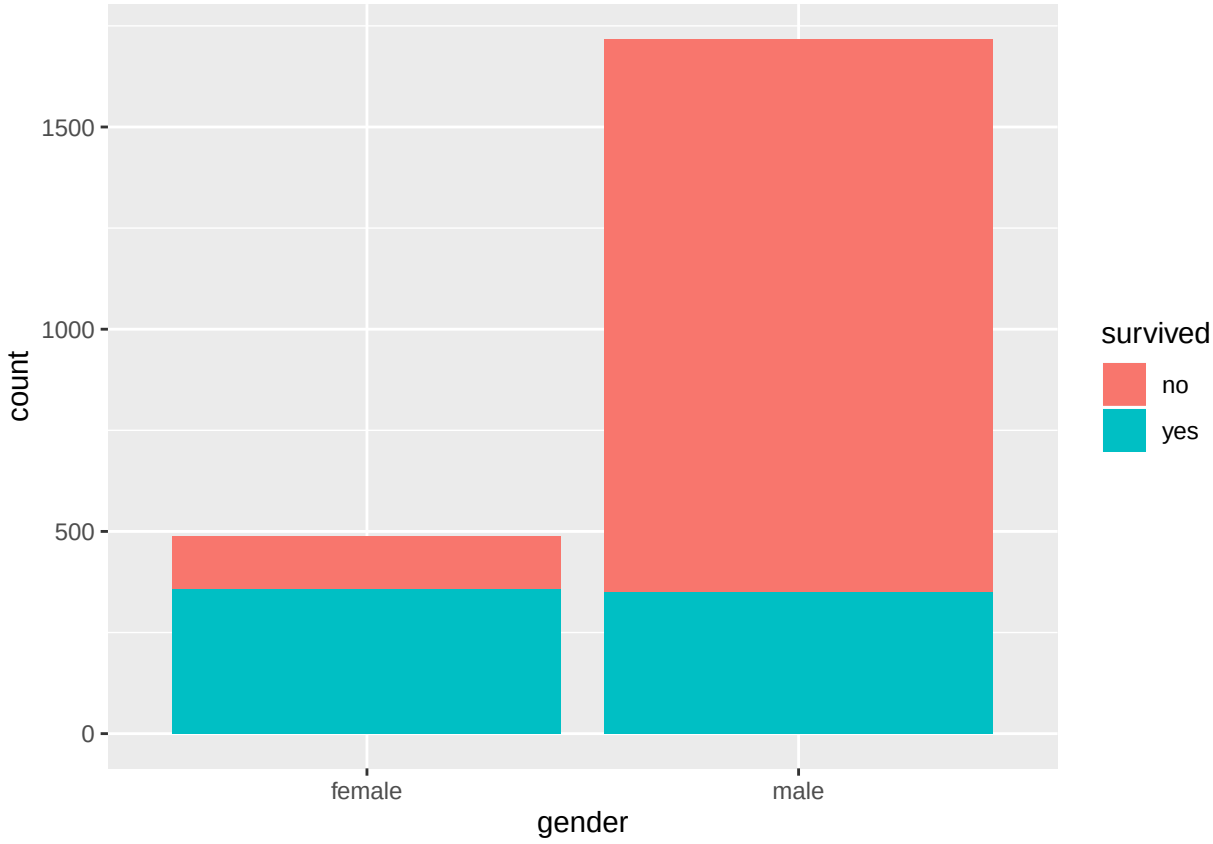
```
# Discretizamos las variables con pocas clases
cols<-c("survived","class","gender","embarked")
for (i in cols){
  totalData[,i] <- as.factor(totalData[,i])
}

# Después de los cambios, analizamos la nueva estructura del conjunto de datos
str(totalData)
```

```
## 'data.frame':    2207 obs. of  13 variables:
##  $ name    : chr  "Abbing, Mr. Anthony" "Abbott, Mr. Eugene Joseph" "Abbott, Mr. Rossmore Edward" "Al
##  $ gender  : Factor w/ 2 levels "female","male": 2 2 2 1 1 2 2 1 2 2 ...
##  $ age     : num  42 13 16 39 16 25 30 28 27 20 ...
##  $ class   : Factor w/ 7 levels "1st","2nd","3rd",..: 3 3 3 3 3 3 2 2 3 3 ...
##  $ embarked: Factor w/ 4 levels "B","C","Q","S": 4 4 4 4 4 4 2 2 2 4 ...
##  $ country : chr  "United States" "United States" "United States" "England" ...
##  $ ticketno: int  5547 2673 2673 2673 348125 348122 3381 3381 2699 3101284 ...
##  $ fare    : num  7.11 20.05 20.05 20.05 7.13 ...
##  $ sibsp   : int  0 0 1 1 0 0 1 1 0 0 ...
##  $ parch   : int  0 2 1 1 0 0 0 0 0 0 ...
##  $ survived: Factor w/ 2 levels "no","yes": 1 1 1 2 2 2 1 2 2 2 ...
##  $ Embarked: chr  NA NA NA NA ...
##  $ Age     : num  NA NA NA NA NA NA NA NA NA NA ...
```

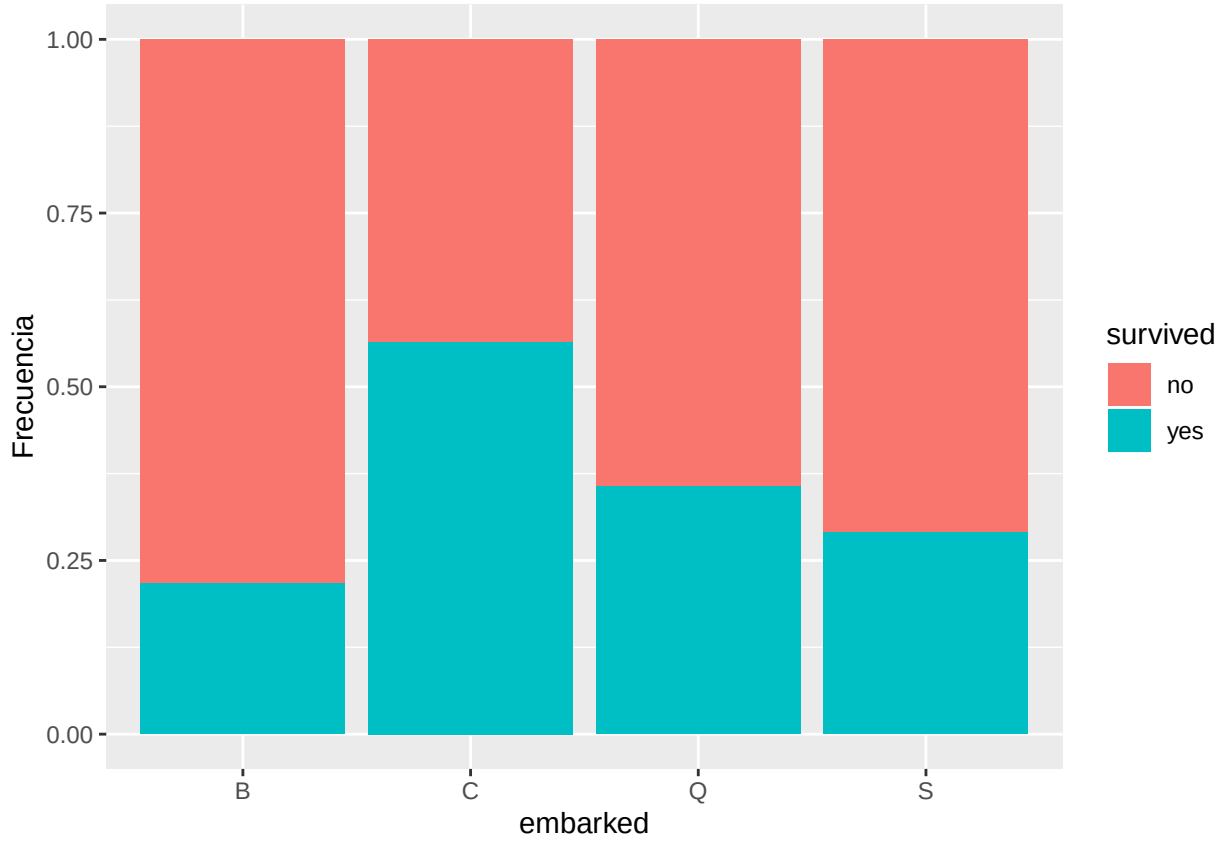
Procesos de análisis del conjunto de datos

Nos proponemos analizar las relaciones entre las diferentes variables del conjunto de datos para ver si se relacionan y como.

```
# Visualizamos la relación entre las variables "sex" y "survival":
ggplot(data=totalData[1:filas,],aes(x=gender,fill=survived))+geom_bar()
```




```
# Otro punto de vista. Survival como función de Embarked:
ggplot(data = totalData[1:filas,],aes(x=embarked,fill=survived))+geom_bar(position="fill")+ylab("Frecuer
```



En la primera gráfica podemos observar fácilmente la cantidad de mujeres que viajaban respecto hombres y observar los que no sobrevivieron. Numéricamente el número de hombres y mujeres supervivientes es similar.

En la segunda gráfica de forma porcentual observamos los puertos de embarque y los porcentajes de supervivencia en función del puerto. Se podría trabajar el puerto C (Cherburgo) para ver de explicar la diferencia en los datos. Quizás porcentualmente embarcaron más mujeres o niños... O gente de primera clase?

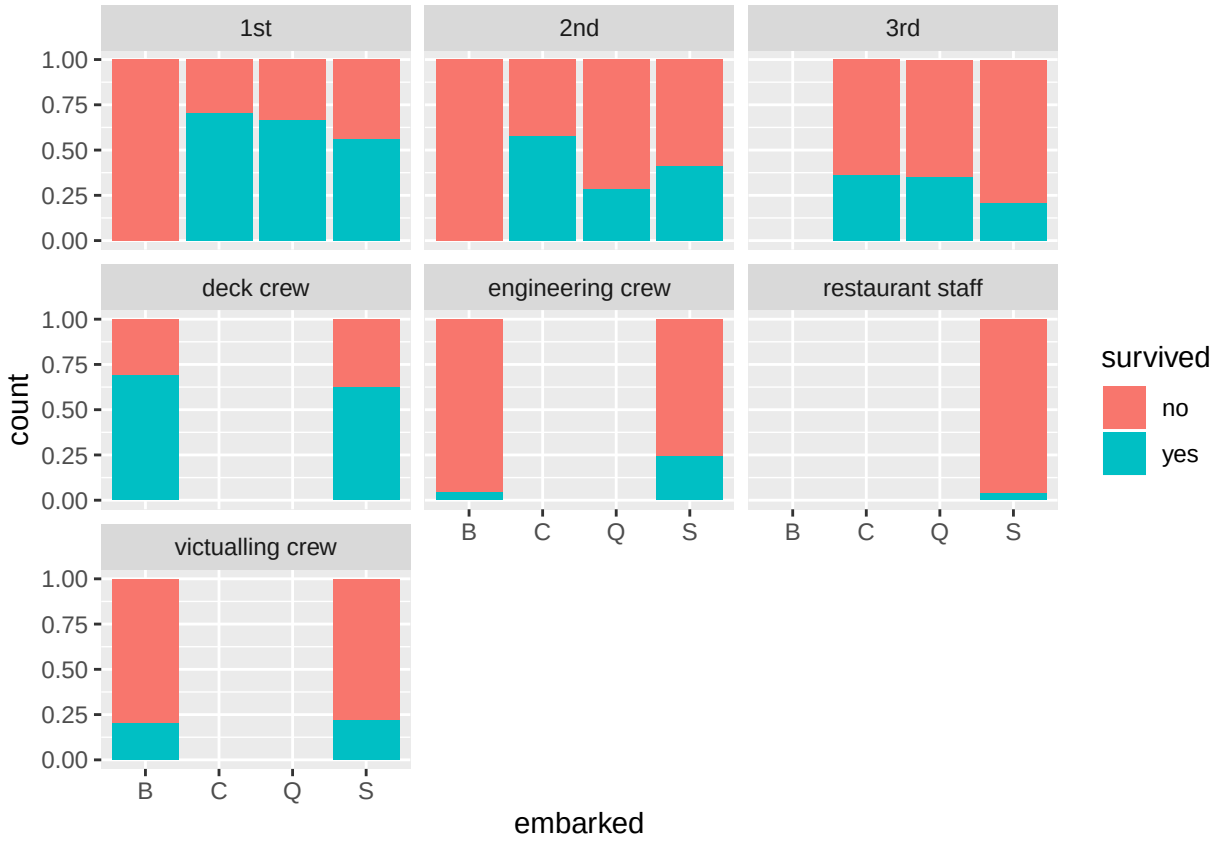
Obtenemos ahora una matriz de porcentajes de frecuencia. Vemos, por ejemplo que la probabilidad de sobrevivir si se embarcó en "C" es de un 56.45%

```
t<-table(totalData[1:filas,]$embarked,totalData[1:filas,]$survived)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##           no      yes
##   B 78.17259 21.82741
##   C 43.54244 56.45756
##   Q 64.22764 35.77236
##   S 70.85396 29.14604
```

Veamos ahora como en un mismo gráfico de frecuencias podemos trabajar con 3 variables: Embarked, Survived y Pclass.

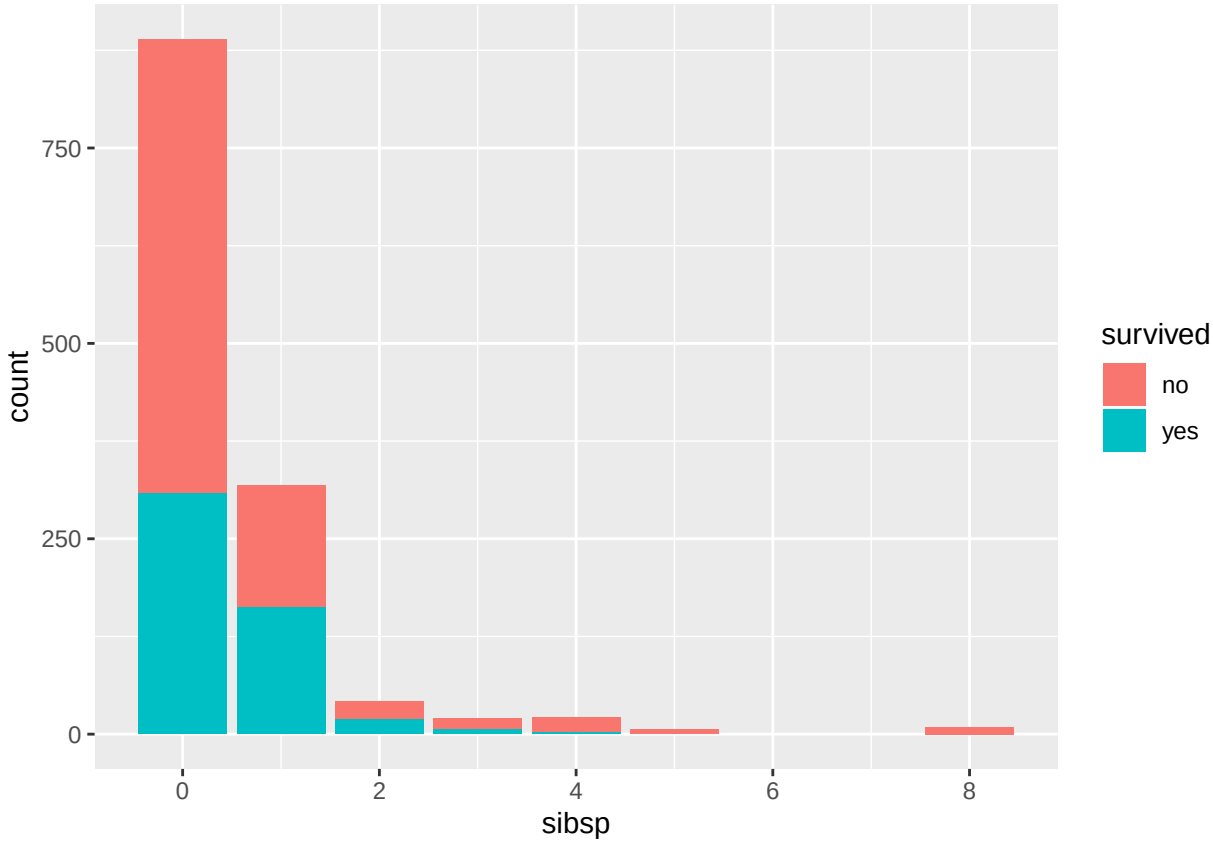
```
# Ahora, podemos dividir el gráfico de Embarked por Pclass:
ggplot(data = totalData[1:filas,],aes(x=embarked,fill=survived))+geom_bar(position="fill")+facet_wrap(~
```



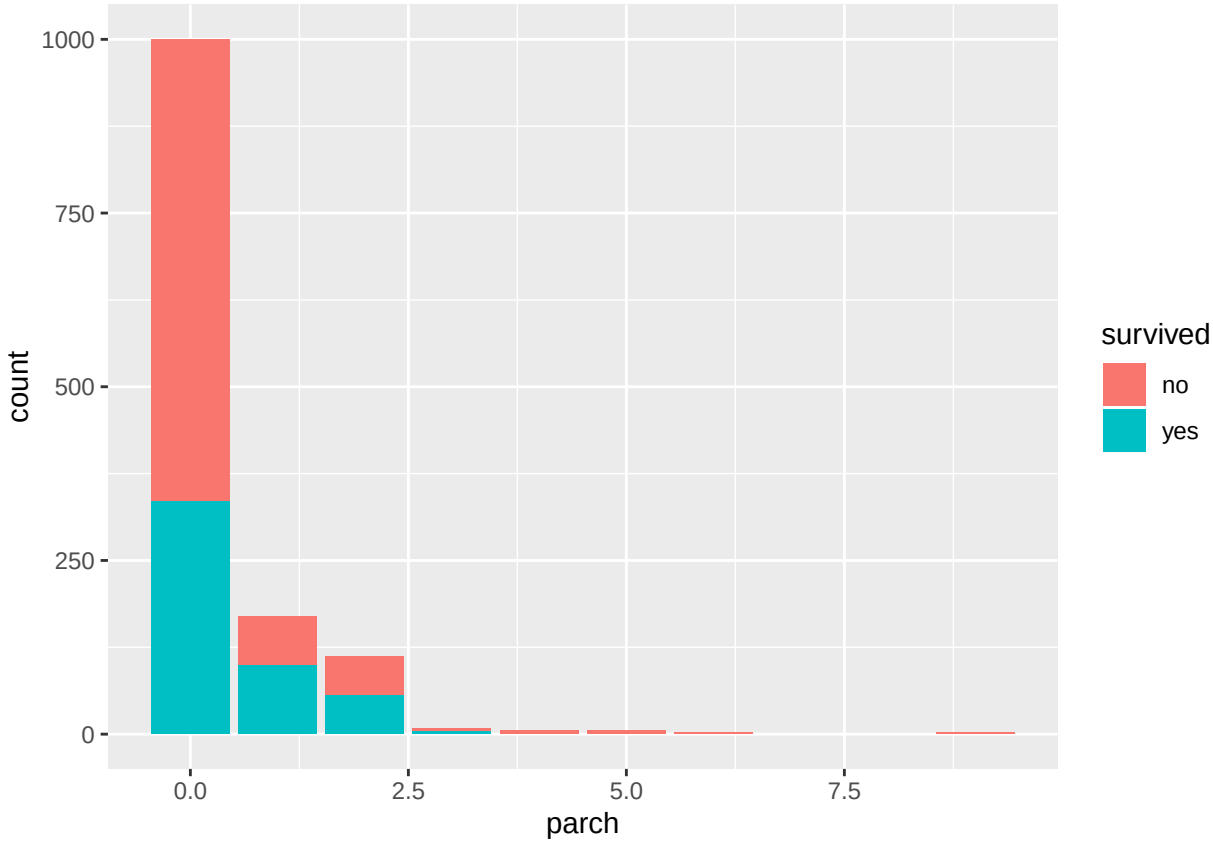
Aquí ya podemos extraer mucha información. Como propuesta de mejora se podría hacer un gráfico similar trabajando solo la clase. Habría que unificar toda la tripulación a una única categoría.

Comparemos ahora dos gráficos de frecuencias: Survived-SibSp y Survived-Parch

```
# Survivial como función de SibSp y Parch
ggplot(data = totalData[1:filas,],aes(x=sibsp,fill=survived))+geom_bar()
```



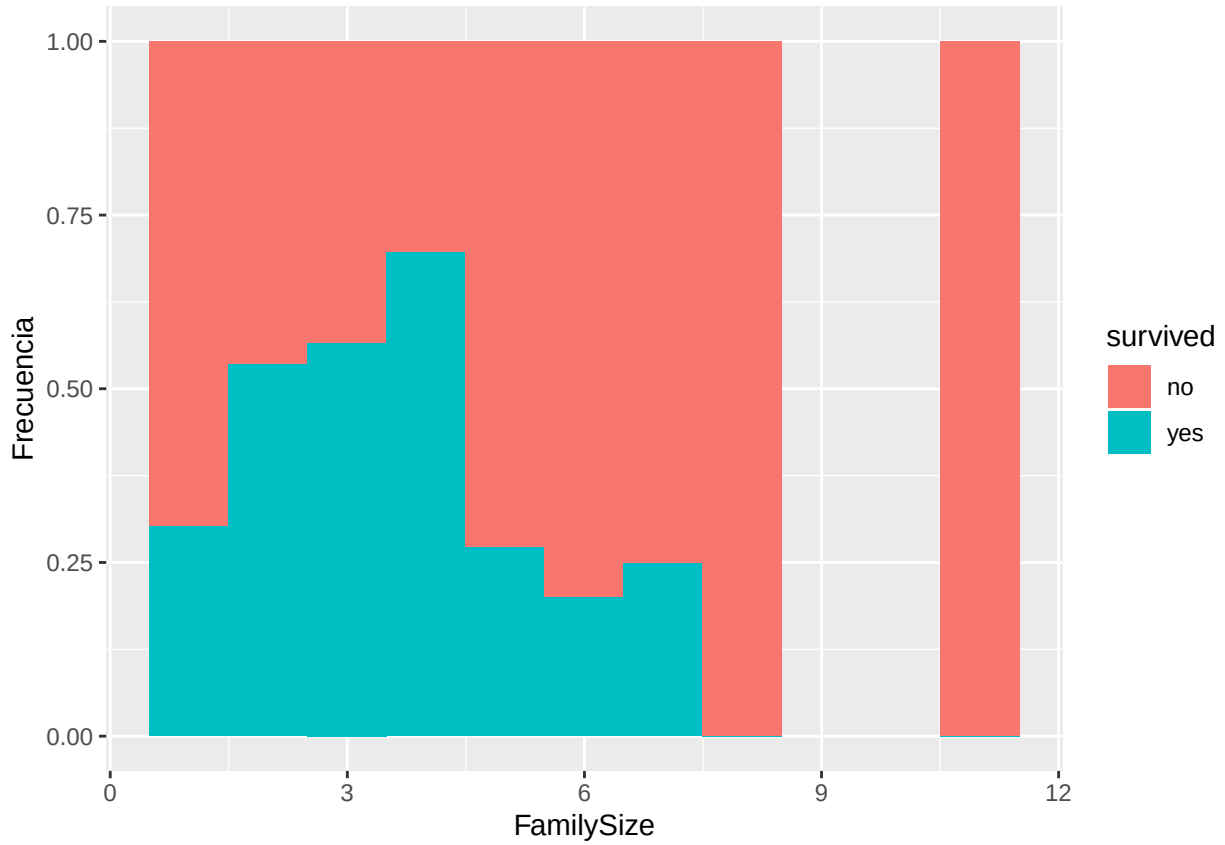

```
ggplot(data = totalData[1:filas,],aes(x=parch,fill=survived))+geom_bar()
```




```
# Vemos como las forma de estos dos gráficos es similar. Este hecho nos puede indicar presencia de corr
```

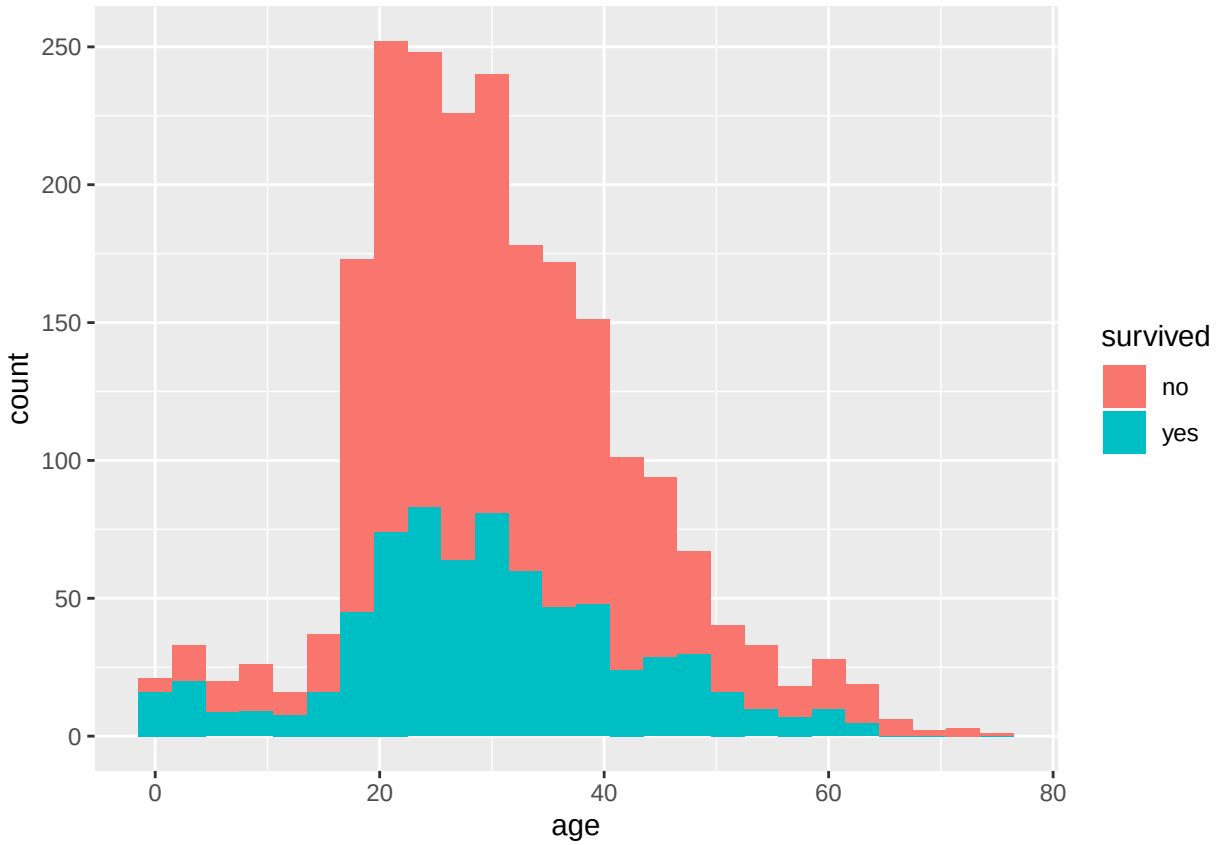
Veamos un ejemplo de construcción de una variable nueva: Tamaño de familia

```
# Construimos un atributo nuevo: family size.
totalData$FamilySize <- totalData$sibsp + totalData$parch +1;
totalData1<-totalData[1:filas,]
ggplot(data = totalData1[!is.na(totalData[1:filas,]$FamilySize),],aes(x=FamilySize,fill=survived))+geom
```

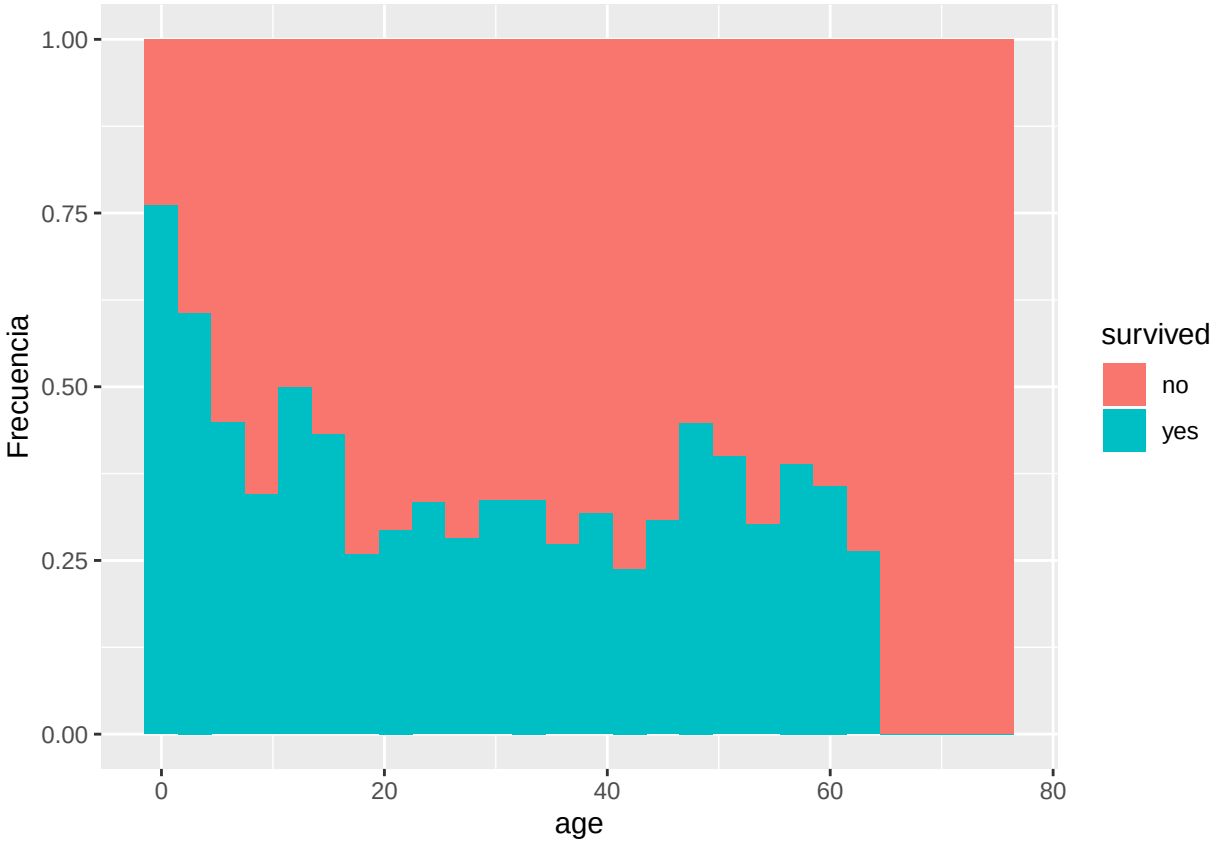


Veamos ahora dos gráficos que nos compara los atributos Age y Survived.
Observamos como el parámetro position="fill" nos da la proporción acumulada de un atributo dentro de otro

```
# Survival como función de age:
ggplot(data = totalData1[!(is.na(totalData[1:filas,]$age)),],aes(x=age,fill=survived))+geom_histogram(bi
```




```
ggplot(data = totalData1[!is.na(totalData[1:filas,]$age),],aes(x=age,fill=survived))+geom_histogram(bin
```



Ejercicios

Ejercicio 1:

Estudia los tres casos siguientes y contesta, de forma razonada la pregunta que se realiza:

- Disponemos de un conjunto de variables referentes a vehículos, tales como la marca, modelo, año de matriculación, etc. También se dispone del precio al que se vendieron. Al poner a la venta a un nuevo vehículo, se dispone de las variables que lo describen, pero se desconoce el precio. ¿Qué tipo de algoritmo se debería aplicar para predecir de forma automática el precio?
- En un almacén de naranjas se tiene una máquina, que de forma automática obtiene un conjunto de variables de cada naranja, como su tamaño, acidez, grado maduración, etc. Si se desea estudiar las naranjas por tipos, según las variables obtenidas, ¿qué tipo de algoritmo es el más adecuado?
- Un servicio de música por internet dispone de los historiales de audición de sus clientes: Qué canciones y qué grupos eligen los clientes a lo largo del tiempo de sus escuchas. La empresa desea crear un sistema que proponga la siguiente canción y grupo en función de la canción que se ha escuchado antes. ¿Qué tipo de algoritmo es el más adecuado?

Respuesta 1:

Escribe aquí la respuesta a la pregunta

Ejercicio 2:

A partir del conjunto de datos disponible en el siguiente enlace http://archive.ics.uci.edu/ml/datasets/Adult , realiza un estudio tomando como propuesta inicial al que se ha realizado con el conjunto de datos "Titanic". Amplia la propuesta generando nuevos indicadores o solucionando otros problemas expuestos en el módulo 2. Explica el proceso que has seguido, qué conocimiento obtienes de los datos, qué objetivo te has fijado y detalla los pasos, técnicas usadas y los problemas resueltos.

Nota: Si lo deseas puedes utilizar otro conjunto de datos propio o de algún repositorio open data siempre que sea similar en diversidad de tipos de variables al propuesto.

Respuesta 2:

####Procesos de limpieza del conjunto de datos.

```
# Cargamos el juego de datos
datosAdult <- read.csv('http://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data',stri
rows=dim(datosAdult)[1]

# Nombres de los atributos
names(datosAdult) <- c("age","workclass","fnlwgt","education","education-num","marital-status","occupati
```

```
# Redacta aquí el código R para el estudio del juego de datos Adult
#Verificamos la estructura del fichero. Observamos que todos los valores tipo string empiezan con un esp
str(datosAdult)
```

```
## 'data.frame':    32561 obs. of  15 variables:
##  $ age           : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ workclass     : chr  " State-gov" " Self-emp-not-inc" " Private" " Private" ...
##  $ fnlwgt        : int  77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 ...
##  $ education     : chr  " Bachelors" " Bachelors" " HS-grad" " 11th" ...
##  $ education-num : int  13 13 9 7 13 14 5 9 14 13 ...
##  $ marital-status: chr  " Never-married" " Married-civ-spouse" " Divorced" " Married-civ-spouse" ...
##  $ occupation    : chr  " Adm-clerical" " Exec-managerial" " Handlers-cleaners" " Handlers-cleaners"
##  $ relationship  : chr  " Not-in-family" " Husband" " Not-in-family" " Husband" ...
##  $ race          : chr  " White" " White" " White" " Black" ...
##  $ sex           : chr  " Male" " Male" " Male" " Male" ...
##  $ capital-gain  : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital-loss  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hour-per-week : int  40 13 40 40 40 40 16 45 50 40 ...
##  $ native-country: chr  " United-States" " United-States" " United-States" " United-States" ...
##  $ income        : chr  " <=50K" " <=50K" " <=50K" " <=50K" ...
```

Descripción de las variables contenidas en el fichero:

-age: a integer with the age of the adults.

-workclass: a string with the workclass of the adults.

-fnlwgt: NI PUTA IDEA: https://www.kaggle.com/uciml/adult-census-income/discussion/32698.

-education: a string wich indicates the education level of the people.

-education-num: a integer with the numbers of years studied per person.

-marital-status: a factor with marital status of the people.

-ocupation: a string with the job of the adults.

-relationship: a string which indicates the civil status.

-race: the race of the adults.

-sex: the sex of the adults.

-capital-gain: the amount of money earned by each adult.

-capital-loss: the amount of money lost by each adult.

-hour-per-week: hours spent per week.

-native-county: country where the adults have born.

-income: amount of money estimated.

Previsualización del csv.

```
head(datosAdult)
```

```
##   age         workclass fnlwgt  education education-num      marital-status
## 1  39         State-gov  77516  Bachelors            13       Never-married
## 2  50  Self-emp-not-inc  83311  Bachelors            13  Married-civ-spouse
## 3  38           Private 215646    HS-grad             9            Divorced
## 4  53           Private 234721       11th             7  Married-civ-spouse
## 5  28           Private 338409  Bachelors            13  Married-civ-spouse
## 6  37           Private 284582    Masters            14  Married-civ-spouse
##           occupation   relationship   race     sex capital-gain capital-loss
## 1       Adm-clerical  Not-in-family  White    Male         2174            0
## 2    Exec-managerial        Husband  White    Male            0            0
## 3  Handlers-cleaners  Not-in-family  White    Male            0            0
## 4  Handlers-cleaners        Husband  Black    Male            0            0
## 5     Prof-specialty           Wife  Black  Female            0            0
## 6    Exec-managerial           Wife  White  Female            0            0
##   hour-per-week native-country income
## 1            40  United-States  <=50K
## 2            13  United-States  <=50K
## 3            40  United-States  <=50K
## 4            40  United-States  <=50K
## 5            40           Cuba  <=50K
## 6            40  United-States  <=50K
```

Mostramos estadísticas bàsicas y después trabajamos los atributos con valores vacíos.

```
#Estadísticas básicas
summary(datosAdult)
```

```
##       age         workclass            fnlwgt         education
##  Min.   :17.00   Length:32561       Min.   :  12285   Length:32561
##  1st Qu.:28.00   Class :character   1st Qu.: 117827   Class :character
##  Median :37.00   Mode  :character   Median : 178356   Mode  :character
##  Mean   :38.58                      Mean   : 189778
##  3rd Qu.:48.00                      3rd Qu.: 237051
##  Max.   :90.00                      Max.   :1484705
##  education-num   marital-status      occupation        relationship
##  Min.   : 1.00   Length:32561       Length:32561       Length:32561
##  1st Qu.: 9.00   Class :character   Class :character   Class :character
##  Median :10.00   Mode  :character   Mode  :character   Mode  :character
##  Mean   :10.08
##  3rd Qu.:12.00
##  Max.   :16.00
##      race               sex             capital-gain    capital-loss
```

```
##  Length:32561       Length:32561       Min.   :    0   Min.   :   0.0
##  Class :character   Class :character   1st Qu.:    0   1st Qu.:   0.0
##  Mode  :character   Mode  :character   Median :    0   Median :   0.0
##                                        Mean   : 1078   Mean   :  87.3
##                                        3rd Qu.:    0   3rd Qu.:   0.0
##                                        Max.   :99999   Max.   :4356.0
##  hour-per-week   native-country        income
##  Min.   : 1.00   Length:32561       Length:32561
##  1st Qu.:40.00   Class :character   Class :character
##  Median :40.00   Mode  :character   Mode  :character
##  Mean   :40.44
##  3rd Qu.:45.00
##  Max.   :99.00
```

```
# Estadísticas de valores vacíos (na)
colSums(is.na(datosAdult))
```

```
##            age      workclass         fnlwgt      education  education-num
##              0              0              0              0              0
## marital-status     occupation   relationship           race            sex
##              0              0              0              0              0
##   capital-gain   capital-loss  hour-per-week native-country         income
##              0              0              0              0              0
```

```
# Estadísticas de valores vacíos
colSums(datosAdult==" ")
```

```
##            age      workclass         fnlwgt      education  education-num
##              0              0              0              0              0
## marital-status     occupation   relationship           race            sex
##              0              0              0              0              0
##   capital-gain   capital-loss  hour-per-week native-country         income
##              0              0              0              0              0
```

```
# Estadísticas de valores vacíos
colSums(datosAdult==" ?")# Esto lo descubrí imprimiendo el dataframe. El print no está en el notebook p
```

```
##            age      workclass         fnlwgt      education  education-num
##              0           1836              0              0              0
## marital-status     occupation   relationship           race            sex
##              0           1843              0              0              0
##   capital-gain   capital-loss  hour-per-week native-country         income
##              0              0              0            583              0
```

```
# Definimos como "Unknown" las variables con valor "?"
datosAdult$workclass[datosAdult$workclass==" ?"]="Unknown"
datosAdult$occupation[datosAdult$occupation==" ?"]="Unknown"
datosAdult$"native-country"[datosAdult$"native-country"==" ?"]="Unknown"
colSums(datosAdult==" ?")
```

```
##            age      workclass         fnlwgt      education  education-num
##              0              0              0              0              0
## marital-status     occupation   relationship           race            sex
##              0              0              0              0              0
##   capital-gain   capital-loss  hour-per-week native-country         income
##              0              0              0              0              0
```

```
#Analizamos qué variables pudieran ser aptas para discretización
apply(datosAdult,2, function(x) length(unique(x)))
```

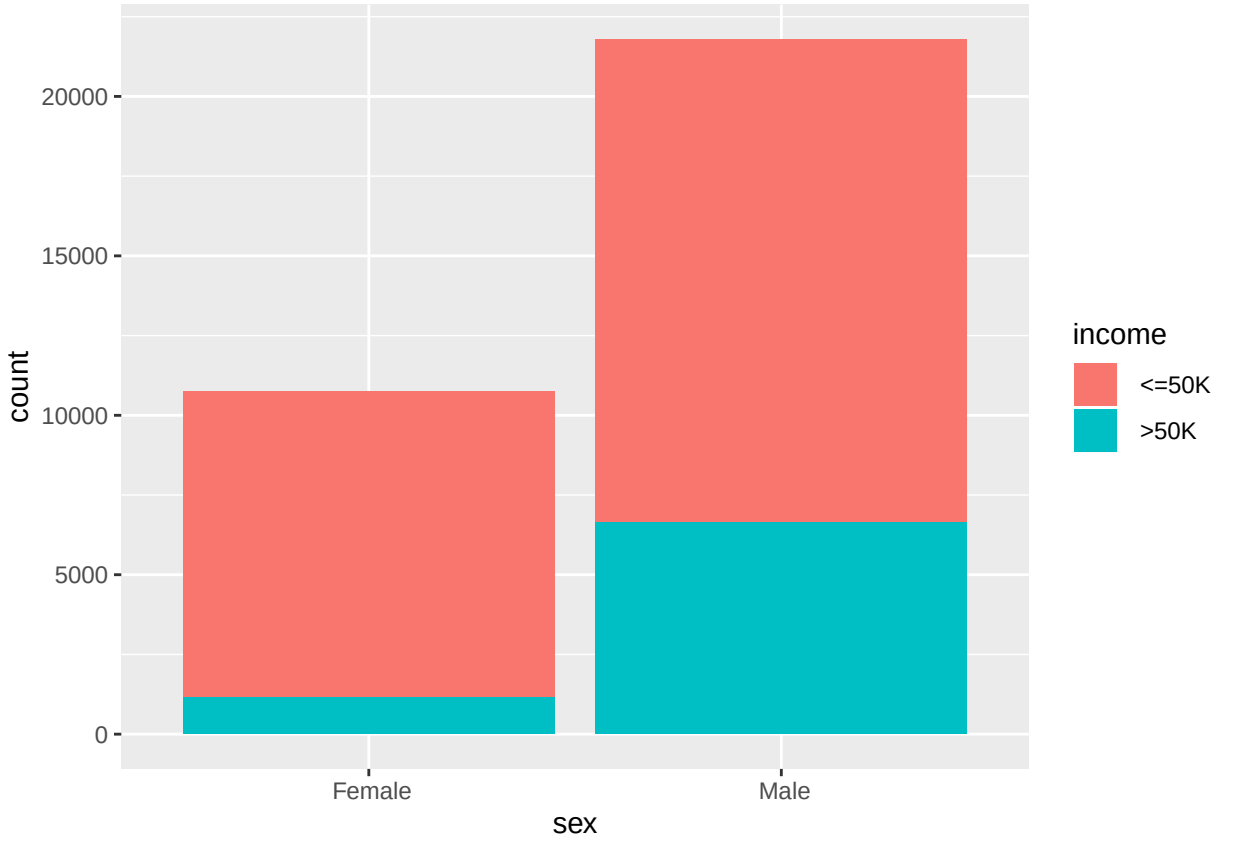
```
##            age      workclass         fnlwgt      education  education-num
##             73              9          21648             16             16
## marital-status     occupation   relationship           race            sex
##              7             15              6              5              2
##   capital-gain   capital-loss  hour-per-week native-country         income
##            119             92             94             42              2
```

```
# Discretizamos las variables con pocas clases y education-num
cols<-c("income","relationship","race","sex", "education-num", "marital-status")
for (i in cols){
  datosAdult[,i] <- as.factor(datosAdult[,i])
  }
str(datosAdult)
```

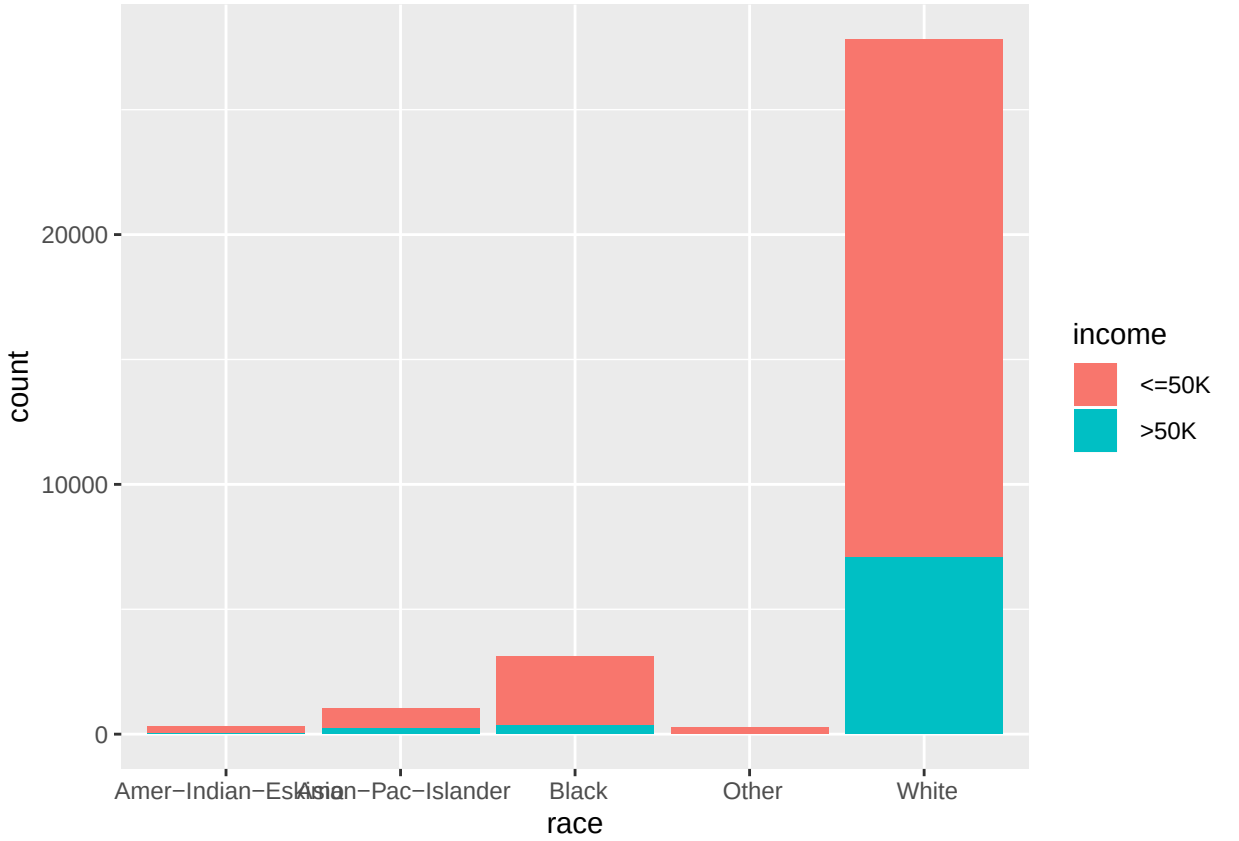
```
## 'data.frame':    32561 obs. of  15 variables:
##  $ age           : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ workclass     : chr  " State-gov" " Self-emp-not-inc" " Private" " Private" ...
##  $ fnlwgt        : int  77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 ...
##  $ education     : chr  " Bachelors" " Bachelors" " HS-grad" " 11th" ...
##  $ education-num : Factor w/ 16 levels "1","2","3","4",..: 13 13 9 7 13 14 5 9 14 13 ...
##  $ marital-status: Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 1 3 3 3 4 3 5 3 ...
##  $ occupation    : chr  " Adm-clerical" " Exec-managerial" " Handlers-cleaners" " Handlers-cleaners"
##  $ relationship  : Factor w/ 6 levels " Husband"," Not-in-family",..: 2 1 2 1 6 6 2 1 2 1 ...
##  $ race          : Factor w/ 5 levels " Amer-Indian-Eskimo",..: 5 5 5 3 3 5 3 5 5 5 ...
##  $ sex           : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 1 1 2 1 2 ...
##  $ capital-gain  : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital-loss  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hour-per-week : int  40 13 40 40 40 40 16 45 50 40 ...
##  $ native-country: chr  " United-States" " United-States" " United-States" " United-States" ...
##  $ income        : Factor w/ 2 levels " <=50K"," >50K": 1 1 1 1 1 1 1 2 2 2 ...
```

Procesos de análisis del conjunto de datos Evaluamos las diferentes relaciones entre los elementos de la población. Se han elegido los que se cree que más información pueden aportar. Posteriormente, se creará un nuevo atributo de datos si es necesario y se implementará un modelo predictivo para income.

```
# Visualizamos la relación entre las variables "sex" y "income":
ggplot(data=datosAdult[1:rows,],aes(x=sex,fill=income))+geom_bar()
```



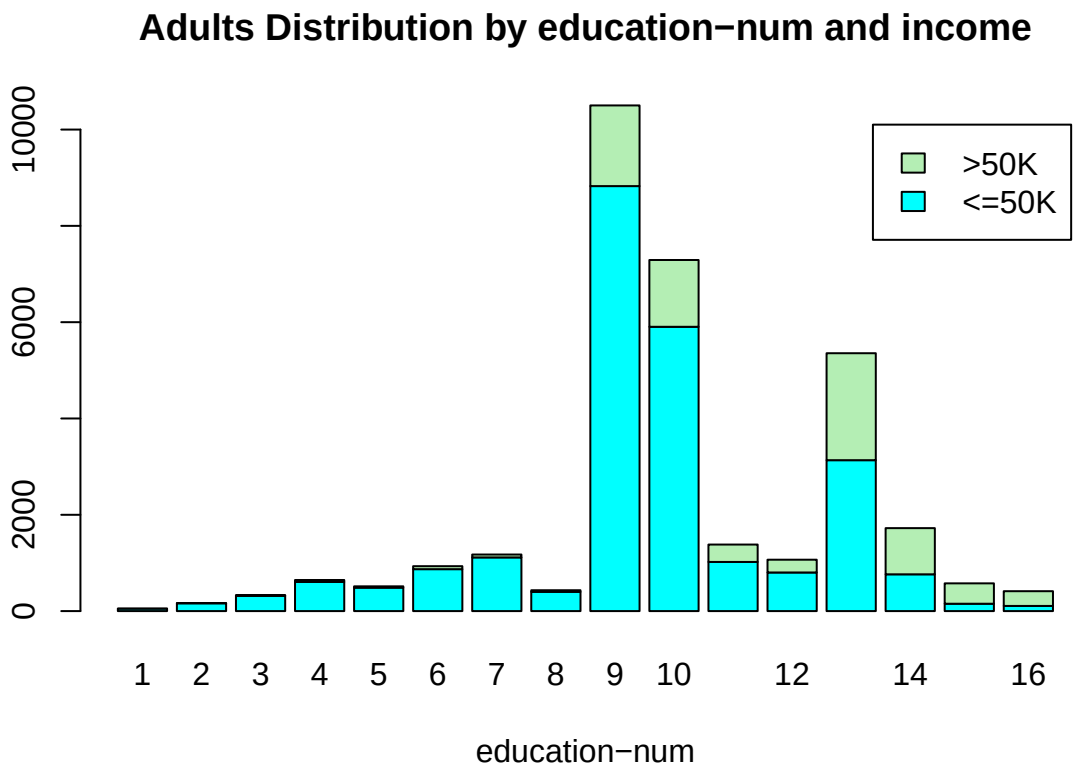

```
# Visualizamos la relación entre las variables "race" y "income":
ggplot(data=datosAdult[1:rows,],aes(x=race,fill=income))+geom_bar()
```




```
t<-table(datosAdult[1:rows,]$"race",datosAdult[1:rows,]$income)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##                          <=50K      >50K
##    Amer-Indian-Eskimo 88.424437 11.575563
##    Asian-Pac-Islander 73.435996 26.564004
##    Black              87.612036 12.387964
##    Other              90.774908  9.225092
##    White              74.414006 25.585994
```

```
# Otro punto de vista. Income como función de education-num:
counts <- table(datosAdult$income, datosAdult$"education-num")
barplot(counts, main="Adults Distribution by education-num and income",
  xlab="education-num",col=c("cyan1","darkseagreen2"),
  legend = rownames(counts))
```



A la hora de analizar los gráficos:

Observamos que ell sexo es relevante para el income. Siendo mucho más igualitaria la proporción en el caso de los hombres.

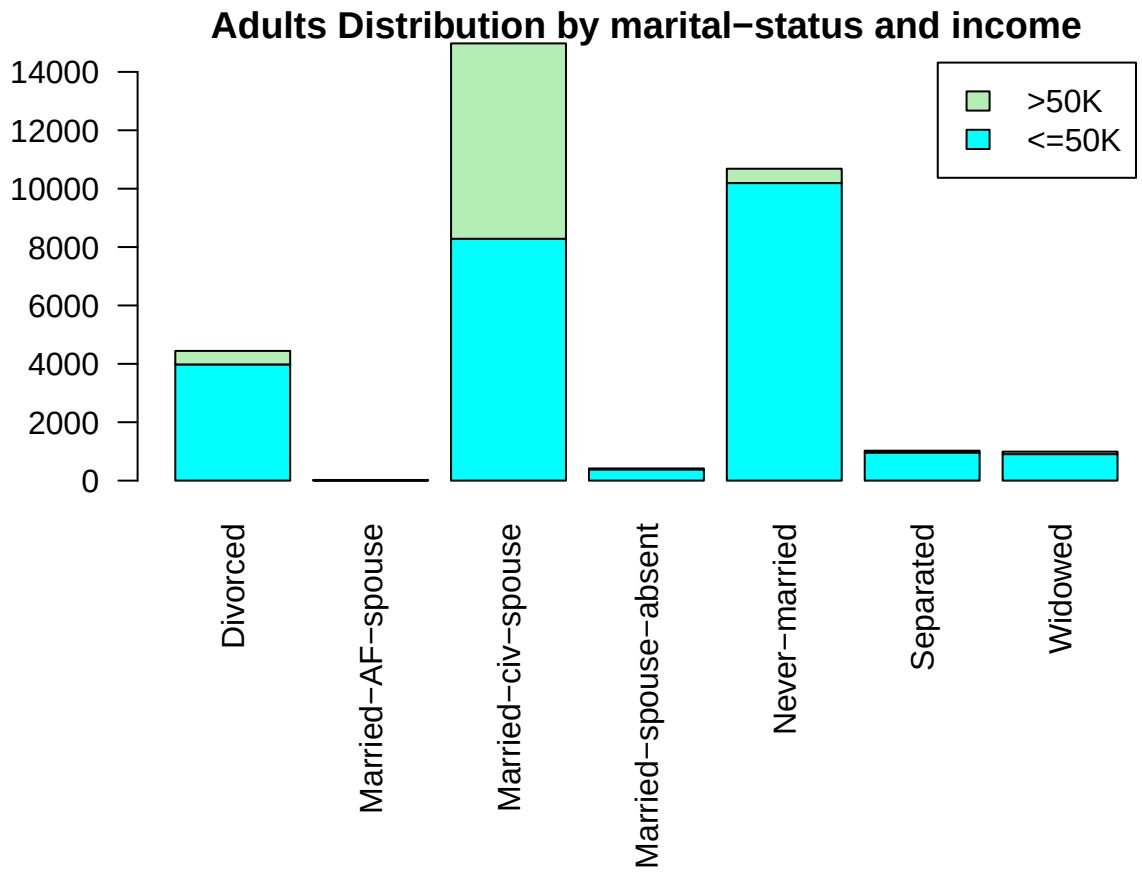
La raza no parece especialmente relevante pero es posible reducir su número clases a 3. Dejando White, black y other.

En cuanto a education-num, se puede ver que por debajo de los 9, la una inmensa mayoría de la población muestra un income de <50k. Es por tanto conveniente colapsar los valores menores que 9 en una única categoría.

```
#Reducción de clases race
new.levels<-c(1,1,2,1,3)
datosAdult$race <- factor(new.levels[datosAdult$race])
#Reducción de clases education-num
new.levels<-c(1,1,1,1,1,1,1,1,2,3,4,5,6,7,8,9)
datosAdult$"education-num" <- factor(new.levels[datosAdult$"education-num"])
```

El estado civil puede ser relevante de cara a los ingresos, si una persona tiene pareja, y ambas trabajan. Es razonable pensar que su "income" será mayor. Factores come éste, están reflejados en el siguiente gráfico.

```
par(mar=c(10,4, 1, 2)) # 15 line height for bottom margin

counts <- table(datosAdult$income, datosAdult$"marital-status")
barplot(counts, main="Adults Distribution by marital-status and income",
  col=c("cyan1","darkseagreen2"),
  legend = rownames(counts), las = 2)
```

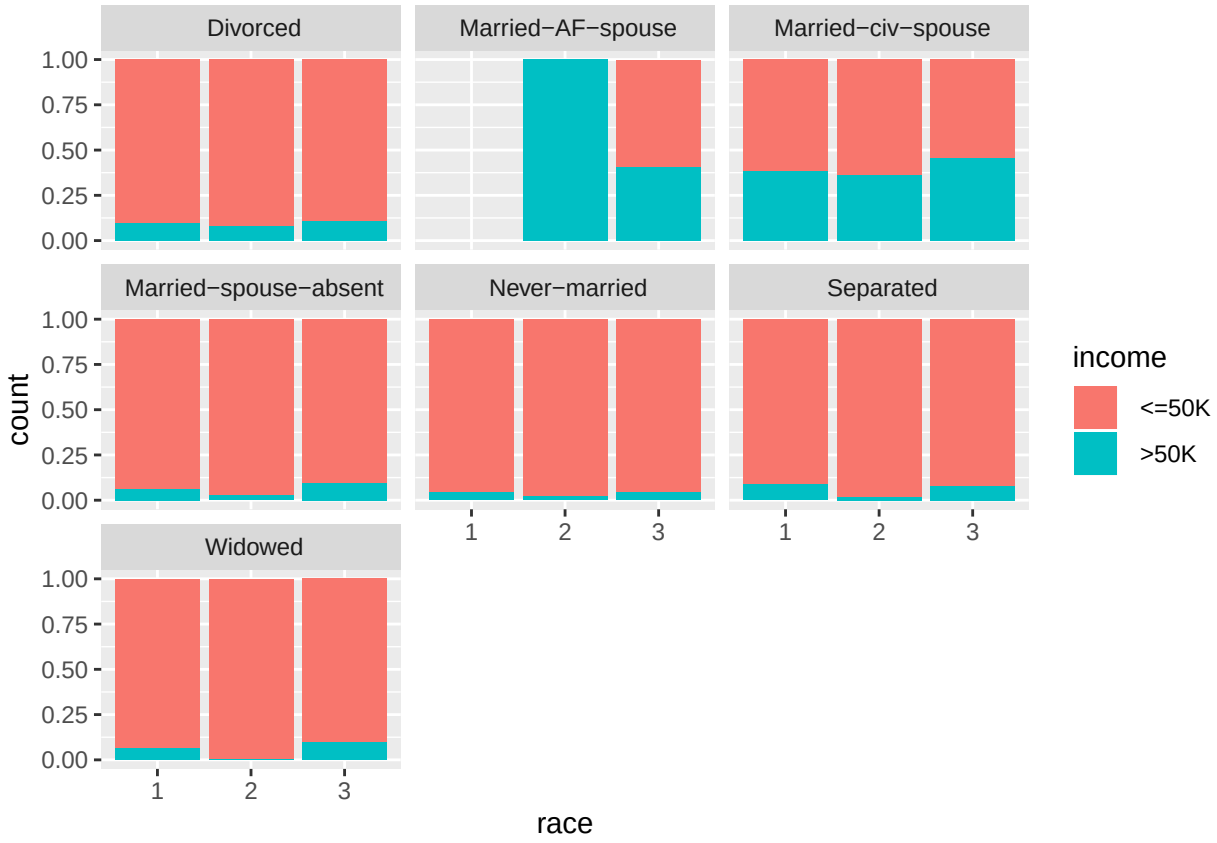


Vemos que el porcentaje de "Married-civ-supouse" con income >50K es sustancialmente mayor que en el resto de las categorías. Obtenemos la matriz de porcentajes para determinar la probabilidad exacta. Es es del 43.47%, mientras que el resto de las categorías tienen cerca de un 90% de probabilidades de tener un income <=50K

```
t<-table(datosAdult[1:rows,]$"marital-status",datosAdult[1:rows,]$income)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##                             <=50K      >50K
##    Divorced              89.579113 10.420887
##    Married-AF-spouse     56.521739 43.478261
##    Married-civ-spouse    55.315171 44.684829
##    Married-spouse-absent 91.866029  8.133971
##    Never-married         95.403913  4.596087
##    Separated             93.560976  6.439024
##    Widowed               91.440081  8.559919
```

```
ggplot(data = datosAdult[1:rows,],aes(x=race,fill=income))+geom_bar(position="fill")+facet_wrap(~datosA
```



Con este gráfico deducimos que las personas de raza negra son siempre las que tienen un porcentaje más elevado. La información a cerca de las personas en "Married-AF-spouse" de raza 1, no es relevante dado que se trata de un número minúsculo de personas. Más hayá de esto el gráfico no revela mayor información relevante. Probamos con el sexo en el lugar de la raza.

```
ggplot(data = datosAdult[1:rows,],aes(x=sex,fill=income))+geom_bar(position="fill")+facet_wrap(~datosAdu
```

Esto nos aporta más información, es curioso ver como la parporción income en el caso de Married-civ-spouse, la clase que más datos alberga, es igual entre distintos sexos, cuando ya hemos visto que como regla genereal esto no sucede con el total de la población.

Resulta evidente que los atributos “capital-gain” y “capital-loss” pueden reducirse estos a un único atributo.

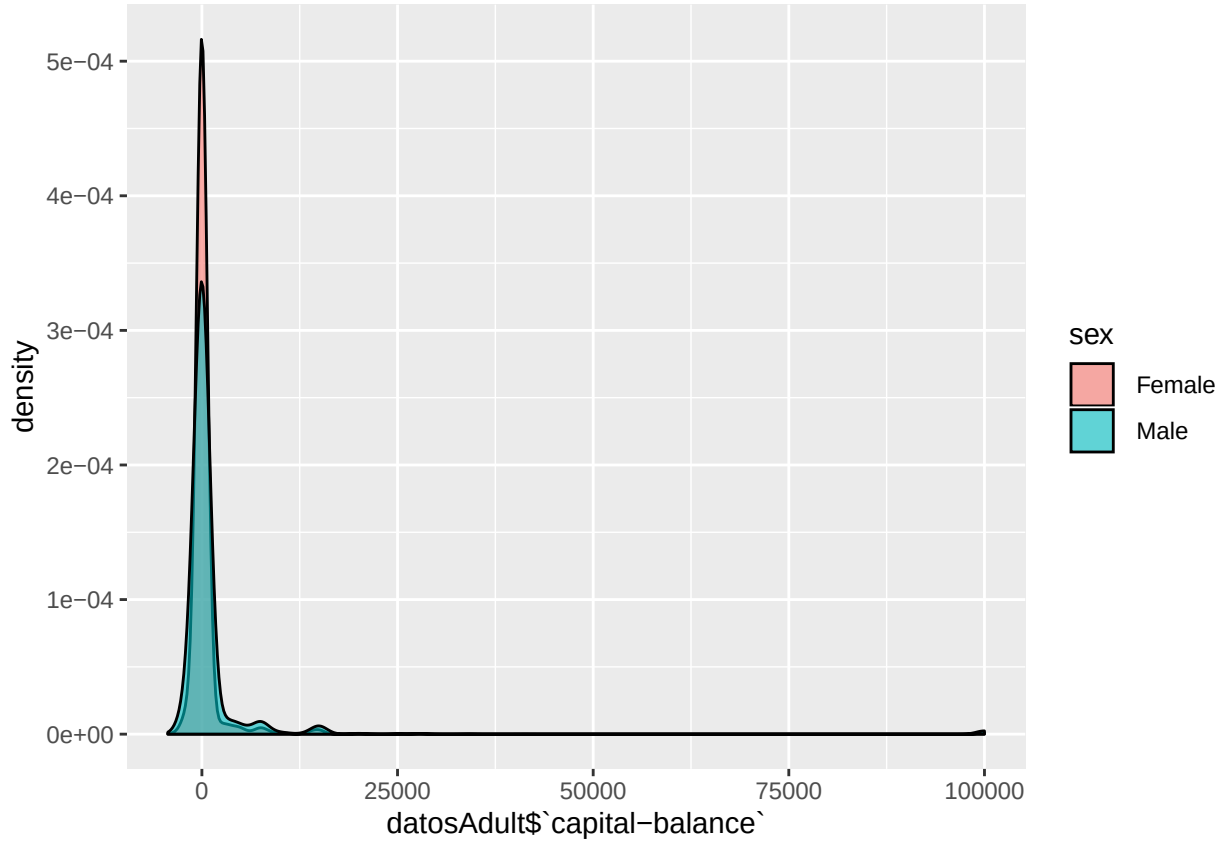
```
datosAdult$`capital-balance`=datosAdult$`capital-gain`-datosAdult$`capital-loss`
all((abs(datosAdult$`capital-balance`)==datosAdult$`capital-gain`+datosAdult$`capital-loss`) ==TRUE)
```

```
## [1] TRUE
```

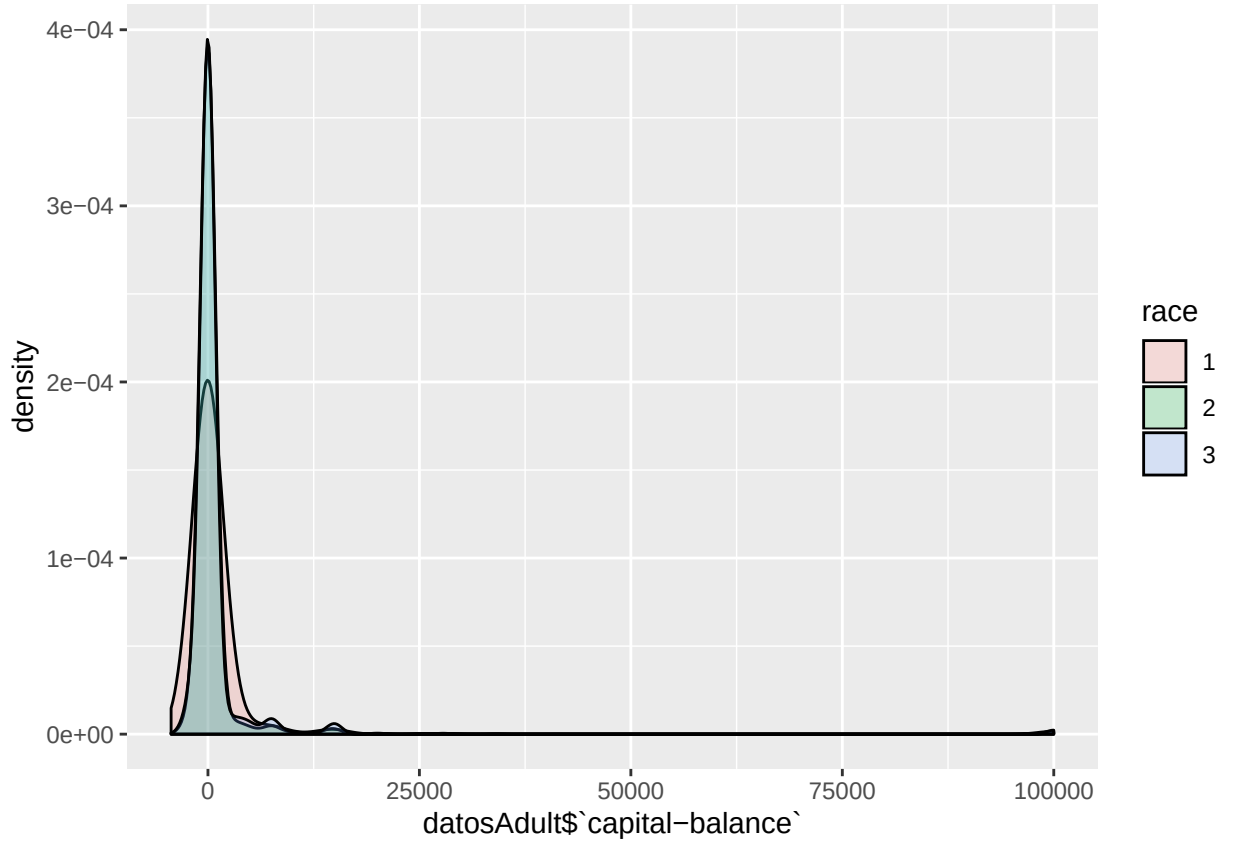
```
# La última línea confirma que si hay "capital-gain", no hay "capital-loss".
```

Ahora miramos el atributo.

```
#En relación con sexo
ggplot(datosAdult, aes(x=datosAdult$`capital-balance`, fill=sex)) +
  geom_density(alpha=0.6)
```




```
#En relación con race
ggplot(datosAdult, aes(x=datosAdult$`capital-balance`, fill=race)) +
  geom_density(alpha=0.2)
```



Fuente para colapsar clases: https://stackoverflow.com/questions/3267312/in-r-how-to-collapse-categories-or-recategorize-variables Fuente de la función barplot(): https://www.rdocumentation.org/packages/graphics/versions/3.6.2/topics/barplot y https://www.r-bloggers.com/setting-graph-margins-in-r-using-the-par-function-and-lots-of-cow-milk/ Fuente de la función all(): https://www.oreilly.com/library/view/the-art-of/9781593273842/ch02s05.html Fuente de la función abs(): https://stackoverflow.com/questions/22306175/change-negative-values-in-dataframe-column-to-absolute-value Fuente de la plot de densidad: http://www.sthda.com/english/wiki/ggplot2-density-plot-quick-start-guide-r-software-and-data-visualization

#####Regresión logísitca

```
input_df <- data.frame('age'=datosAdult$age,'sex'=datosAdult$sex,'relationship'=datosAdult$relationship
                       'education-num'=datosAdult$`education-num`,'marital-status'=datosAdult$`marital-
                       'capital-balance'=datosAdult$`capital-balance`,'income'=datosAdult$income)

cols<-c("sex","race","income","relationship","race","sex", "education-num", "marital-status")
for (i in cols){
  datosAdult[,i] <- as.factor(datosAdult[,i])
  }
str(input_df)
```

```
## 'data.frame':    32561 obs. of  8 variables:
##  $ age            : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ sex            : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 1 1 2 1 2 ...
##  $ relationship   : Factor w/ 6 levels " Husband"," Not-in-family",..: 2 1 2 1 6 6 2 1 2 1 ...
##  $ race           : Factor w/ 3 levels "1","2","3": 3 3 3 2 2 3 2 3 3 3 ...
##  $ education.num  : Factor w/ 9 levels "1","2","3","4",..: 6 6 2 1 6 7 1 2 7 6 ...
##  $ marital.status : Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 1 3 3 3 4 3 5 3 ...
```

```
##  $ capital.balance: int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ income         : Factor w/ 2 levels " <=50K"," >50K": 1 1 1 1 1 1 1 2 2 2 ...
```

```
#Normalización min-max
normalize <- function(x) {
return ((x - min(x)) / (max(x) - min(x))+1)
}
input_df$age<-normalize(input_df$age)
input_df$capital.balance<-normalize(input_df$age)
str(input_df)
```

```
## 'data.frame':    32561 obs. of  8 variables:
##  $ age            : num  1.3 1.45 1.29 1.49 1.15 ...
##  $ sex            : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 1 1 2 1 2 ...
##  $ relationship   : Factor w/ 6 levels " Husband"," Not-in-family",..: 2 1 2 1 6 6 2 1 2 1 ...
##  $ race           : Factor w/ 3 levels "1","2","3": 3 3 3 2 2 3 2 3 3 3 ...
##  $ education.num  : Factor w/ 9 levels "1","2","3","4",..: 6 6 2 1 6 7 1 2 7 6 ...
##  $ marital.status : Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 1 3 3 3 4 3 5 3 ...
##  $ capital.balance: num  1.3 1.45 1.29 1.49 1.15 ...
##  $ income         : Factor w/ 2 levels " <=50K"," >50K": 1 1 1 1 1 1 1 2 2 2 ...
```

```
#Split the data
library(caret)
```

```
## Loading required package: lattice
```

```
inTrain <- createDataPartition(y = input_df$income, p = .60, list = FALSE)
training <- input_df[inTrain,]
testing <- input_df[-inTrain,]
dim(training)
```

```
## [1] 19537     8
```

```
dim(testing)
```

```
## [1] 13024     8
```

```
input_df.fit = glm(income ~ age + sex + relationship + race + education.num + marital.status + capital.
summary(input_df.fit)
```

```
##
## Call:
## glm(formula = income ~ age + sex + relationship + race + education.num +
##     marital.status + capital.balance, family = binomial, data = training)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3691  -0.5940  -0.2413  -0.0450   3.5650
##
## Coefficients: (1 not defined because of singularities)
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                        -7.90396    0.40676 -19.432  < 2e-16 ***
## age                                 1.65980    0.13393  12.393  < 2e-16 ***
## sex Male                            0.90625    0.08982  10.090  < 2e-16 ***
## relationship Not-in-family          0.79128    0.30081   2.631  0.00853 **
## relationship Other-relative        -0.33096    0.27840  -1.189  0.23452
## relationship Own-child             -0.53123    0.29766  -1.785  0.07431 .
## relationship Unmarried              0.61341    0.32006   1.917  0.05530 .
```

```
## relationship Wife                      1.06785    0.11827   9.029  < 2e-16 ***
## race2                                  0.24936    0.13240   1.883  0.05964 .
## race3                                  0.47587    0.10410   4.571 4.85e-06 ***
## education.num2                         1.20059    0.09778  12.278  < 2e-16 ***
## education.num3                         1.76359    0.10085  17.487  < 2e-16 ***
## education.num4                         1.99251    0.12741  15.639  < 2e-16 ***
## education.num5                         1.94599    0.14019  13.881  < 2e-16 ***
## education.num6                         2.81352    0.10138  27.753  < 2e-16 ***
## education.num7                         3.33286    0.11879  28.057  < 2e-16 ***
## education.num8                         3.80771    0.17016  22.378  < 2e-16 ***
## education.num9                         4.09381    0.19461  21.036  < 2e-16 ***
## marital.status Married-AF-spouse       2.96783    0.61588   4.819 1.44e-06 ***
## marital.status Married-civ-spouse      2.26182    0.30255   7.476 7.66e-14 ***
## marital.status Married-spouse-absent -0.63727    0.28771  -2.215  0.02676 *
## marital.status Never-married         -0.62002    0.09993  -6.205 5.48e-10 ***
## marital.status Separated             -0.26764    0.19658  -1.361  0.17336
## marital.status Widowed               -0.23910    0.17940  -1.333  0.18259
## capital.balance                            NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 21570  on 19536  degrees of freedom
## Residual deviance: 14417  on 19513  degrees of freedom
## AIC: 14465
##
## Number of Fisher Scoring iterations: 7
```

```
#Prediction with using a threshold of 0.5
input_df.prob = predict(input_df.fit, testing, type="response")
input_df.pred = rep("<=50K", dim(training)[1])
input_df.pred[input_df.prob > .5] = ">50K"
table(input_df.pred, training$income)
```

```
##
## input_df.pred  <=50K  >50K
##         <=50K  12319  3926
##         >50K    2513   779
```

```
total=((13630+371)/(13630+371+1202+4334))
cat("Acc",total)
```

```
## Acc 0.7166402
```

Fuente de ejercicios de regresión logística: http://rstudio-pubs-static.s3.amazonaws.com/74431_8cbd662559f6451f9cd411545f28107f.html https://stats.idre.ucla.edu/r/dae/logit-regression/ Fuente de la función createDataPartition(): https://www.rdocumentation.org/packages/caret/versions/6.0-85/topics/createDataPartition
Fuente de la función glm(): https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/glm
Fuente de la función cat(): https://stackoverflow.com/questions/15589601/print-string-and-variable-contents-on-the-same-line-in-r

---

Rúbrica

Pregunta Concepto Peso en la nota final

1ª Se acierta al identificar el tipo de problema que presenta el caso. 5%

1ª La explicación proporcionada es correcta. La justificación y argumentación está suficientemente elaborada. 5%

1b Se acierta al identificar el tipo de problema que presenta el caso. 5%

1b La explicación proporcionada es correcta. La justificación y argumentación está suficientemente elaborada. 5%

1c Se acierta al identificar el tipo de problema que presenta el caso. 5%

1c La explicación proporcionada es correcta. La justificación y argumentación está suficientemente elaborada. 5%

2 Se carga la base de datos, se visualiza su estructura y se explican los hechos básicos. 5%

2 Se estudia si existen atributos vacíos, y si es el caso, se adoptan medidas para tratar estos atributos. 2.5%

2 Se transforma algún atributo para adaptarlo en un estudio posterior. 2.5%

2 Se realiza alguna discretització de algún atributo. 5%

2 Se crea un indicador nuevo a partido otros atributos 5%

2 Se analizan los datos de forma visual y se extraen conclusiones tangibles. Hay que elaborar un discurso coherente y con conclusiones claras. 35%

2 Se trata en profundidad algún otro aspecto respecto a los datos presentado en el módulo 2 10%

2 Se ha buscado información adicional, se ha incluido en el documento de respuesta y las fuentes se han citado correctamente 5%