Limpieza y análisis de datos

Adrià Cortés Andrés - MPilar García Ruiz

12/12/2021

0.- Preliminares

Esta práctica se ha realizado bajo el contexto de la asignatura Tipología y ciclo de vida de los datos (M2.851), perteneciente al Máster en Ciencia de Datos de la Universitat Oberta de Catalunya (UOC). En ella, se aplican técnicas de limpieza y análisis de datos mediante el lenguaje de programación R con el fin de limpiar y analizar un conjunto de datos para después estudiar la probabilidad de alcanzar (o no) cierto umbral de salario en base a los factores personales descritos en el dataset.

Miembros del equipo El proyecto ha sido realizado de forma conjunta por:

- Adrià Cortés Andrés
- María Pilar Garcia Ruiz

1.- Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende responder?

El conjunto de datos ha sido obtenido de la plataforma kaggle, en la url

[Kaggle: Your Machine Learning and Data Science Community] https://www.kaggle.com/ddmasterdon/income-adult?select=adult_data.csv

Las variables con las que cuenta el dataset son las siguientes:

Variables	Tipo	Descripción
Age	Numérico	Edad de la persona
workclass	Texto	Clase de empleado. Campo categorizado.
fnlwgt	Numérico	Peso final dado por la oficina que ha recogido los datos y que da el número de unidades en la población objetivo en base a una fórmula no facilitada-
education	Texto	Nivel de estudios. Campo categorizado
education-num	Texto	Años dedicados a los estudios
marital-status	Texto	Situación sentimental. Campo categorizado
occupation	Texto	Trabajo actual
relationship	Texto	Relación familiar actual. Campo categorizado
race	Texto	Raza. Campo categorizado
sex	Texto	Sexo. Campo categorizado
capital-gain	Numérico	Ganancia de capital

Variables	Tipo	Descripción
capital-loss	Numérico	Pérdida de capital
hours-per-week	Numérico	Horas de trabajo por semana
native-country	Texto	Nacionalidad. Campo categorizado
salary	Lógico	Salario

Inicialmente se escoge este dataset para ver si se puede estudiar qué salario puede alcanzar una persona en base a su pertenencia a un grupo demográfico, con las características descritas en el conjunto de datos.
Las personas que no alcanzan un determinado umbral de sueldo tienen una peor calidad de vida, se encuentran con más dificultades para hacer frente a cualquier eventualidad negativa y esto provoca otras consecuencias (problemas de adicción, riesgos de problemas de salud, tanto físicos como mentales, etcétera). Determinados grupos demográficos pueden convertirse en colectivos más vulnerables de la sociedad. Factores como la raza, el sexo o el nivel de estudios, influyen en el acceso a la vida laboral y pueden ser determinantes en el momento de conseguir una mejor situación laboral.

2.- Integración y selección de los datos de interés a analizar.

Se inicia la práctica con la lectura del fichero de datos. Para ello, cargaremos los paquetes necesarios, cuyo número puede irse incrementando conforme avance el desarrollo de la práctica.

Paquetes utilizados Paquetes utilizados en esta práctica:

• dplyr (manejo de dataframes) • rmarkdown (informe dinámico) • nortest (test lillie) • ggplot2 (gráficas) • gridExtra (formato gráficas) • mlbench (regresión logística) • Corrplot (correlation matrix) • Tidyr (ordenación de datos) • Caret (métodos de entrenamiento y clasificación) • gbm (regresión logística)

Lectura del fichero Se realiza la lectura con la función read.csv, indicando la coma como carácter separador y diciéndole que contamos con una primera fila de cabecera de columnas.

```
# fichero a cargar: ./data/adult_data.csv
datos <- read.csv (file.choose(),
                   sep = ",",
                   header = TRUE)

unmodifiedDatos <- datos
```

```
dim (datos)
```

Dimensión del conjunto de datos

```
## [1] 32561    15
```

```
str (datos)
```

Características de las variables leídas

```
## '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 ...
##  $ hours.per.week: int  40 13 40 40 40 40 16 45 50 40 ...
##  $ native.country: chr  " United-States" " United-States" " United-States" " United-States" ...
##  $ salary        : chr  " <=50K" " <=50K" " <=50K" " <=50K" ...
```

Para seguir con el análisis preliminar, podemos, por ejemplo, hacer una inspección de 10 observaciones elegidas de manera aleatoria como ejemplo para ilustrar los datos con los que contamos. Aunque dichos valores ya se muestran en el estudio de la estructura, visualizar el dataset en columnas siempre es más claro.

```
set.seed (1)
datos[sample (nrow (datos)
              , 10),
      ]
```

```
##       age workclass fnlwgt     education education.num      marital.status
## 17401  51   Private 441637       HS-grad             9  Married-civ-spouse
## 24388  38   Private 115289       HS-grad             9  Married-civ-spouse
## 4775   30   Private  53158    Assoc-acdm            12       Never-married
## 26753  38   Private 172538     Bachelors            13  Married-civ-spouse
## 13218  21   Private 200973  Some-college            10       Never-married
## 26109  34   Private  80058   Prof-school            15       Never-married
## 29143  33   Private  90668       HS-grad             9       Never-married
## 10539  66   Private 185336       HS-grad             9             Widowed
## 8462   38   Private 108140     Bachelors            13       Never-married
## 4050   28   Private  25955     Assoc-voc            11            Divorced
##                 occupation    relationship                race     sex capital.gain
## 17401       Tech-support         Husband               White    Male            0
## 24388  Machine-op-inspct         Husband               White    Male            0
## 4775        Tech-support   Not-in-family               White  Female            0
## 26753       Tech-support         Husband               White    Male            0
## 13218              Sales       Own-child               White  Female            0
## 26109    Exec-managerial       Own-child               White    Male            0
## 29143       Adm-clerical   Not-in-family               White  Female            0
## 10539              Sales  Other-relative               White  Female            0
## 8462       Other-service   Not-in-family               White    Male            0
## 4050        Craft-repair   Not-in-family  Amer-Indian-Eskimo    Male            0
##       capital.loss hours.per.week native.country salary
## 17401            0             40  United-States  <=50K
## 24388            0             40  United-States  <=50K
## 4775             0             40  United-States  <=50K
## 26753         1977             40  United-States   >50K
## 13218            0             20  United-States  <=50K
## 26109            0             50  United-States  <=50K
## 29143            0             40  United-States  <=50K
## 10539            0             35  United-States  <=50K
## 8462             0             20  United-States  <=50K
## 4050             0             40  United-States  <=50K
```

Dado el estudio que se pretende hacer, y tras el estudio de las variables, se desestiman para la práctica la ganancia y pérdida de capital.

```
datos <- dplyr::select (datos,
                        -capital.gain,
                        -capital.loss)

str (datos)
```

```
## 'data.frame':    32561 obs. of  13 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" ...
##  $ hours.per.week: int  40 13 40 40 40 40 16 45 50 40 ...
##  $ native.country: chr  " United-States" " United-States" " United-States" " United-States" ...
##  $ salary        : chr  " <=50K" " <=50K" " <=50K" " <=50K" ...
```

Seguimos con la inclusión de una variable secuencial como identificador único de cada fila. Además, realizamos una primera reordenación de las columnas.

```
filas <- nrow (datos)

datos <- cbind (datos,
                'id' = 1:filas)

datos <- subset (datos,
                 select = c(14,1,2,3,4,5,6,7,8,9,10,11,12,13))

names (datos)
```

```
##  [1] "id"             "age"            "workclass"      "fnlwgt"         "education"
##  [6] "education.num"  "marital.status" "occupation"     "relationship"   "race"
## [11] "sex"            "hours.per.week" "native.country" "salary"
```

La segmentación por rangos de edad permite establecer nichos más concretos y a profundizar en el estudio. Este procedimiento, conocido como discretización (transformar variables numéricas en variables categóricas) ayuda a hacer el dataset más agradable, y lo hace más fácil de analizar y extraer conclusiones. Se establecen categorías de edad para el estudio de empleabilidad vs salario en las siguientes escalas:

- 0 a 16 años, cat_0
- 17 a 20, cat_1
- 21 a 34, cat_2
- 35 a 45, cat_3
- 46 a 65, cat_4
- 65 ~, cat_5

```
datos <- cbind (datos,
                age_cat = cut (datos$age,
                               breaks = c(0,16,20,34,45,65,150),
                               labels = c("cat_0",
                                          "cat_1",
                                          "cat_2",
                                          "cat_3",
                                          "cat_4",
                                          "cat_5"),
                               right = TRUE))


names (datos)
```

```
##  [1] "id"             "age"            "workclass"      "fnlwgt"         "education"
##  [6] "education.num"  "marital.status" "occupation"     "relationship"   "race"
## [11] "sex"            "hours.per.week" "native.country" "salary"         "age_cat"
```

Preprocesado - Limpieza de espacios en blanco en las variables Se cambian las variables que empiezan por un espacio. Se crea una función para ello.

```
ltrim <- function (x) {trimws(x,which = c("left"))}

datos$workclass <- ltrim (datos$workclass)
datos$education <- ltrim (datos$education)
datos$marital.status <- ltrim (datos$marital.status)
datos$occupation <- ltrim (datos$occupation)
datos$relationship <- ltrim (datos$relationship)
datos$race <- ltrim (datos$race)
datos$sex <- ltrim (datos$sex)
datos$salary <- ltrim (datos$salary)
```

Cambiar los niveles de ingresos a un valor numérico de 0 o 1 para el modelado de clasificación, incluida la regresión logística

```
datos$salary = gsub("<=50K",
                    0,
                    datos$salary)

datos$salary = gsub(">50K",
                    1,
                    datos$salary)
```

Cambiamos el tipo de datos a numérico.

```
datos$salary <- as.numeric(datos$salary)
class (datos$salary)
```

```
## [1] "numeric"
```

Preprocesado - Variables categóricas Podemos combinar algunos valores categóricos para reducir la dispersión y facilitar los análisis. Especialmente útil en las variables en las que el reparto de las cuales no ayude o no sea relevante en los análisis, como en el caso del estado civil, dónde la diferencia entre separado y divorciado no aporta nada.

Marital.status

```
datos$marital.status [datos$marital.status == "Never-married"] = "Not-Married"
datos$marital.status [datos$marital.status == "Married-AF-spouse"] = "Married"
datos$marital.status [datos$marital.status == "Married-civ-spouse"] = "Married"
datos$marital.status [datos$marital.status == "Married-spouse-absent"] = "Not-Married"
datos$marital.status [datos$marital.status == "Separated"] = "Not-Married"
datos$marital.status [datos$marital.status == "Divorced"] = "Not-Married"
datos$marital.status [datos$marital.status == "Widowed"] = "Widowed"
```

workclass

```
datos$workclass = gsub ("^Federal-gov",
                        "Federal-Govt",
                        datos$workclass)
datos$workclass = gsub ("^Local-gov",
                        "Other-Govt",
                        datos$workclass)
datos$workclass = gsub ("^State-gov",
                        "Other-Govt",
                        datos$workclass)
datos$workclass = gsub ("^Private",
                        "Private",
                        datos$workclass)
datos$workclass = gsub ("^Self-emp-inc",
                        "Self-Employed",
                        datos$workclass)
datos$workclass = gsub ("^Self-emp-not-inc",
                        "Self-Employed",
                        datos$workclass)
datos$workclass = gsub ("^Without-pay",
                        "Not-Working",
                        datos$workclass)
datos$workclass = gsub ("^Never-worked",
                        "Not-Working",
                        datos$workclass)
```

occupation

```
datos$occupation = gsub ("^Adm-clerical",
                         "Admin",
                         datos$occupation)
datos$occupation = gsub ("^Armed-Forces",
                         "Military",
                         datos$occupation)
datos$occupation = gsub ("^Craft-repair",
                         "Blue-Collar",
                         datos$occupation)
datos$occupation = gsub ("^Exec-managerial",
                         "White-Collar",
                         datos$occupation)
datos$occupation = gsub ("^Farming-fishing",
                         "Blue-Collar",
                         datos$occupation)
datos$occupation = gsub ("^Handlers-cleaners",
                         "Blue-Collar",
                         datos$occupation)
```

```
datos$occupation = gsub ("^Machine-op-inspct",
                         "Blue-Collar",
                         datos$occupation)
datos$occupation = gsub ("^Other-service",
                         "Service",
                         datos$occupation)
datos$occupation = gsub ("^Priv-house-serv",
                         "Service",
                         datos$occupation)
datos$occupation = gsub ("^Prof-specialty",
                         "Professional",
                         datos$occupation)
datos$occupation = gsub ("^Protective-serv",
                         "Other-Occupations",
                         datos$occupation)
datos$occupation = gsub ("^Sales",
                         "Sales",
                         datos$occupation)
datos$occupation = gsub ("^Tech-support",
                         "Other-Occupations",
                         datos$occupation)
datos$occupation = gsub ("^Transport-moving",
                         "Blue-Collar",
                         datos$occupation)
```

3.- Limpieza de los datos.

3.1.- ¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos? Tras un análisis visual, y utilizando rstudio, se localizan datos perdidos. A continuación, pasamos a la revisión de las variables para su análisis y limpieza.

```
table (datos$age)
```

```
##
##  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38
## 395 550 712 753 720 765 877 798 841 785 835 867 813 861 888 828 875 886 876 898 858 827
##  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60
## 816 794 808 780 770 724 734 737 708 543 577 602 595 478 464 415 419 366 358 366 355 312
##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82
## 300 258 230 208 178 150 151 120 108  89  72  67  64  51  45  46  29  23  22  22  20  12
##  83  84  85  86  87  88  90
##   6  10   3   1   1   3  43
```

```
table (datos$workclass)
```

```
##
##             ?  Federal-Govt   Not-Working     Other-Govt       Private Self-Employed
##          1836           960            21          3391         22696          3657
```

```
table (datos$education)
```

```
##
##         10th         11th         12th      1st-4th      5th-6th      7th-8th
##          933         1175          433          168          333          646
##          9th   Assoc-acdm    Assoc-voc    Bachelors    Doctorate      HS-grad
##          514         1067         1382         5355          413        10501
```

```
##      Masters    Preschool  Prof-school Some-college
##         1723           51          576         7291
```

```
table (datos$education.num)
```

```
##
##     1     2     3     4     5     6     7     8     9    10    11    12    13    14
##    51   168   333   646   514   933  1175   433 10501  7291  1382  1067  5355  1723
##    15    16
##   576   413
```

```
table (datos$marital.status)
```

```
##
##     Married Not-Married     Widowed
##       14999       16569         993
```

```
table (datos$occupation)
```

```
##
##                 ?             Admin       Blue-Collar          Military
##              1843              3770             10062                 9
## Other-Occupations      Professional             Sales           Service
##              1577              4140              3650              3444
##      White-Collar
##              4066
```

```
table (datos$relationship)
```

```
##
##        Husband  Not-in-family Other-relative      Own-child      Unmarried
##          13193           8305            981           5068           3446
##           Wife
##           1568
```

```
table (datos$race)
```

```
##
## Amer-Indian-Eskimo Asian-Pac-Islander              Black              Other
##                311               1039               3124                271
##              White
##              27816
```

```
table (datos$sex)
```

```
##
## Female   Male
##  10771  21790
```

```
table (datos$hours.per.week)
```

```
##
##     1     2     3     4     5     6     7     8     9    10    11    12    13    14
##    20    32    39    54    60    64    26   145    18   278    11   173    23    34
##    15    16    17    18    19    20    21    22    23    24    25    26    27    28
##   404   205    29    75    14  1224    24    44    21   252   674    30    30    86
##    29    30    31    32    33    34    35    36    37    38    39    40    41    42
##     7  1149     5   266    39    28  1297   220   149   476    38 15217    36   219
##    43    44    45    46    47    48    49    50    51    52    53    54    55    56
```

```
##   151   212  1824    82    49   517    29  2819    13   138    25    41   694    97
##    57    58    59    60    61    62    63    64    65    66    67    68    70    72
##    17    28     5  1475     2    18    10    14   244    17     4    12   291    71
##    73    74    75    76    77    78    80    81    82    84    85    86    87    88
##     2     1    66     3     6     8   133     3     1    45    13     2     1     2
##    89    90    91    92    94    95    96    97    98    99
##     2    29     3     1     1     2     5     2    11    85
```

```
table (datos$native.country)
```

```
##
##                          ?                   Cambodia                     Canada
##                        583                         19                        121
##                      China                   Columbia                       Cuba
##                         75                         59                         95
##           Dominican-Republic                    Ecuador                El-Salvador
##                         70                         28                        106
##                    England                     France                    Germany
##                         90                         29                        137
##                     Greece                  Guatemala                      Haiti
##                         29                         64                         44
##           Holand-Netherlands                   Honduras                       Hong
##                          1                         13                         20
##                    Hungary                      India                       Iran
##                         13                        100                         43
##                    Ireland                      Italy                    Jamaica
##                         24                         73                         81
##                      Japan                        Laos                     Mexico
##                         62                         18                        643
##                  Nicaragua  Outlying-US(Guam-USVI-etc)                       Peru
##                         34                         14                         31
##                Philippines                     Poland                   Portugal
##                        198                         60                         37
##                Puerto-Rico                   Scotland                      South
##                        114                         12                         80
##                     Taiwan                   Thailand           Trinadad&Tobago
##                         51                         18                         19
##              United-States                    Vietnam                 Yugoslavia
##                      29170                         67                         16
```

```
table (datos$salary)
```

```
##
##     0     1
## 24720  7841
```

Tras el análisis de las distintas variables, hemos encontrado valores '?', que consideramos como perdidos, en las siguientes variables

- workclass
- occupation
- native.country

Procedemos a sustituirlos.

```
datos [datos == "?"] <- NA
```

Del total de datos leídos, los valores perdidos son

```
sum (is.na (datos))
```

```
## [1] 3679
```

y por columna

```
apply (is.na (datos),
       2,
       sum)
```

```
##             id            age      workclass         fnlwgt      education
##              0              0           1836              0              0
##  education.num marital.status     occupation   relationship           race
##              0              0           1843              0              0
##            sex hours.per.week native.country         salary        age_cat
##              0              0              0              0              0
```

La media de valores perdidos por columna sería

```
apply (is.na (datos),
       2,
       mean)
```

```
##             id            age      workclass         fnlwgt      education
##     0.00000000     0.00000000     0.05638647     0.00000000     0.00000000
##  education.num marital.status     occupation   relationship           race
##     0.00000000     0.00000000     0.05660146     0.00000000     0.00000000
##            sex hours.per.week native.country         salary        age_cat
##     0.00000000     0.00000000     0.00000000     0.00000000     0.00000000
```

Creamos una categoría Unknown en Native Country.

```
datos$native.country [is.na (datos$native.country)] <- "Unknown"
```

Para las variables workclass y occupation consideramos que son despreciables en el volumen total de datos, por lo que podemos eliminarlos.

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

Con lo que nuestro dataframe ha cambiado de dimensión.

```
dim (datos)
```

```
## [1] 30718    15
```

Guardamos en un fichero los datos tratados hasta este punto.

3.2.- Identificación y tratamiento de valores extremos. Gráficamente

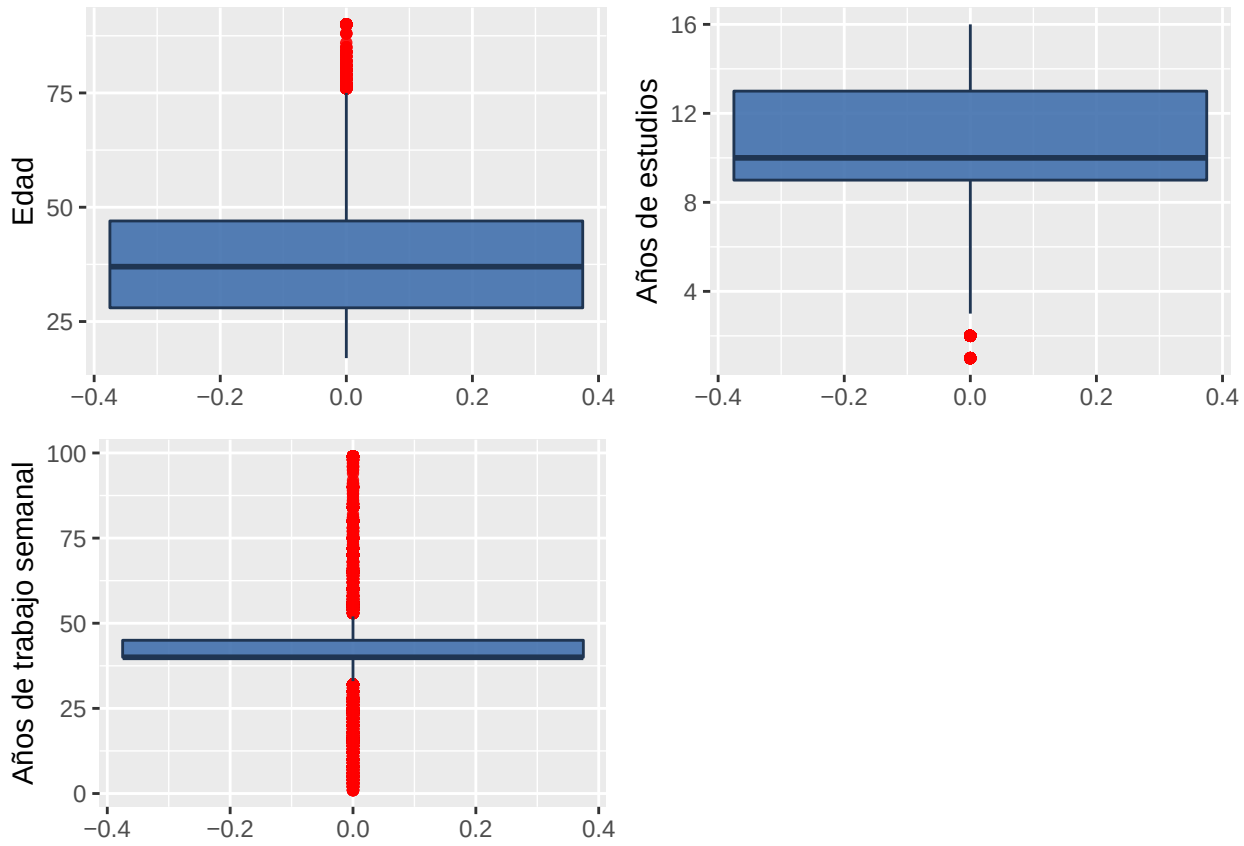
```
par (mfrow = c(2,3))
#Edad
plotV1 <- ggplot2::ggplot (data = datos,
                  aes (y = age)) +
  geom_boxplot (fill = "#4271AE",
                colour = "#1F3552",
                alpha = 0.9,
                outlier.colour = "red") +
```

```
  scale_y_continuous (name = "Edad")

#Años de estudios
plotV2 <- ggplot2::ggplot (data = datos,
                  aes (y = education.num)) +
  geom_boxplot (fill = "#4271AE",
                colour = "#1F3552",
                alpha = 0.9,
                outlier.colour = "red") +
  scale_y_continuous (name = "Años de estudios")

#Horas de trabajo
plotV3 <- ggplot2::ggplot (data = datos,
                  aes (y = hours.per.week)) +
  geom_boxplot (fill = "#4271AE",
                colour = "#1F3552",
                alpha = 0.9,
                outlier.colour = "red") +
  scale_y_continuous (name = "Años de trabajo semanal")

grid.arrange (plotV1,
              plotV2,
              plotV3,
              ncol=2)
```



Por funciones, que nos muestren el valor de los outliers por variable

```
boxplot.stats (datos$age)$out
```

```
##   [1] 79 76 90 77 76 81 78 90 88 90 77 90 77 78 80
##  [16] 90 81 81 76 80 90 76 79 76 81 76 90 76 90 80
##  [31] 90 90 79 78 79 84 90 77 80 77 90 81 83 84 79
##  [46] 76 85 82 79 77 90 76 90 84 78 78 76 80 90 90
##  [61] 77 76 84 76 90 76 90 76 77 81 90 77 78 77 81
##  [76] 78 82 81 77 76 80 90 80 84 82 78 79 76 90 84
##  [91] 90 83 78 80 77 78 76 79 80 79 80 90 90 90 90
## [106] 81 76 83 90 90 81 80 80 90 79 77 77 80 76 82
## [121] 85 80 79 90 76 76 77 76 79 81 77 88 90 82 76
## [136] 88 76 77 83 76 77 79 77 86 90 77 82 83 81 76
## [151] 79 76 84 78 76 76 76 78 84 79 78 90 80 81 78
## [166] 81 90 80 82 90 90 85
```

Tras estudiar algunas de las variables numéricas, se encuentran muchos valores que la función boxplots.stats devuelve como extremos.

```
boxplot.stats (datos$education.num)$out
```

```
##   [1] 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 1 2 1 2 1 2
##  [23] 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 1
##  [45] 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 1 2
##  [67] 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2
##  [89] 1 2 2 2 2 2 1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2
## [111] 2 2 2 1 2 2 2 1 1 2 2 1 1 2 2 1 2 2 1 2 1 2
## [133] 2 2 2 2 1 2 1 1 2 1 1 1 2 1 2 1 2 2 1 1 2 2
## [155] 2 2 1 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2
## [177] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 1 2
## [199] 1 2 2 1
```

Nos quedaríamos con el estudio de la variable age, donde efectivamente los valores extremos se corresponden con edades en las que se podría asumir que la persona ya no está en trabajo activo. El estudio podría establecer no tener en cuenta las filas de estos individuos.

4.- Análisis de los datos.

4.1.- Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).

Normalidad El estudio que interesa es la relación de las distintas variables con el umbral de salario especificado. Así, a priori, se debería realizar análisis de las variables:

- age_cat~salary. Cobran más las categorías de edad más elevada que los más jóvenes.
- sex~salary. Cobran más los hombres que las mujeres.
- race~salary. Cobran más los individuos de raza blanca que los de otras razas.

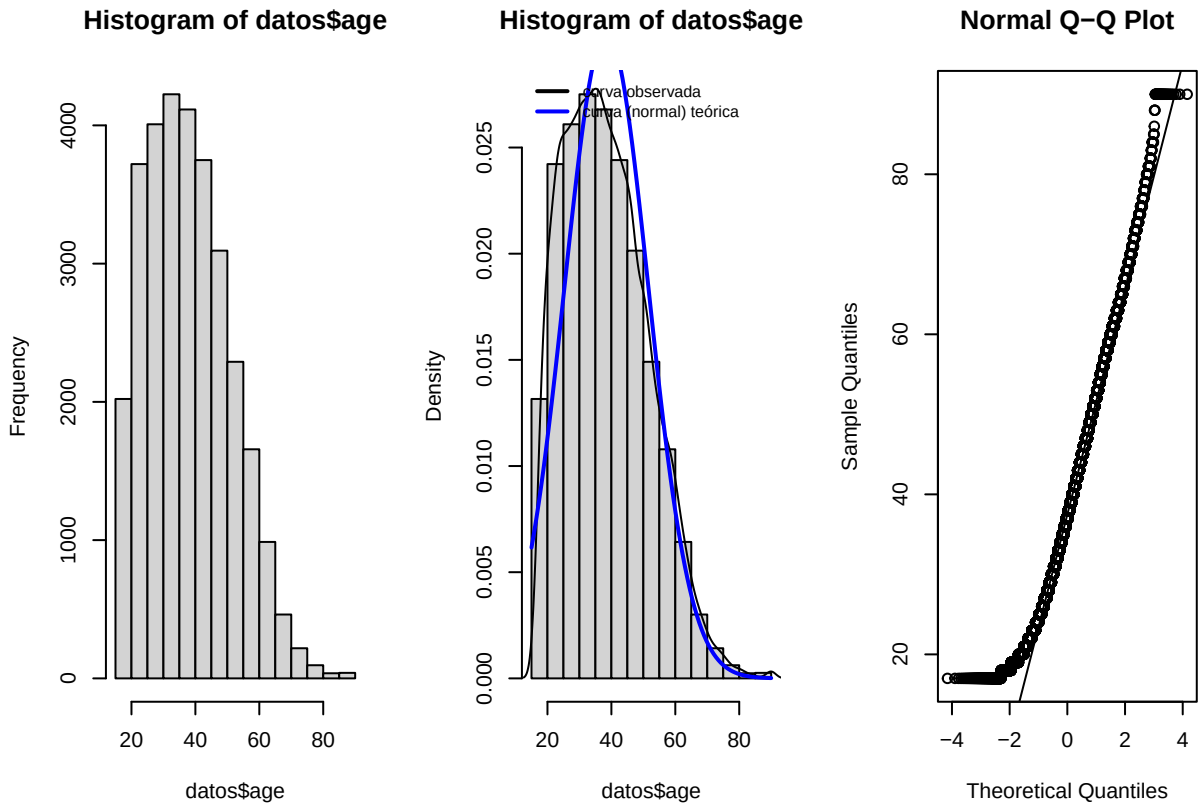
4.2.- Comprobación de la normalidad y homogeneidad de la varianza. Dado el elevado número de datos, podría aplicarse el teorema del límite central. También puede usarse uno de los tests de los disponibles en r. Dado el elevado tamaño de la muestra se usará lillie.test. Si el valor de probabilidad (p-value) que obtenemos por la prueba es menor a 0.05 se rechaza la hipótesis nula y los datos no siguen una distribución normal. Si el valor de probabilidad es mayor a 0.05, no se rechazaría la hipótesis nula y los datos seguirían una distribución normal.

```
nortest::lillie.test (datos$age)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality
##  test
##
## data:  datos$age
## D = 0.060447, p-value < 2.2e-16
```

El valor p-value del test de Lilliefors es mucho menor al 0.05 por tanto se rechaza la hipótesis nula y se concluye que la variable edad no sigue una distribución normal. Visualmente quedaría

```
par (mfrow = c(1,3))
hist (datos$age)
hist (datos$age,
      freq=F)
lines (density (datos$age))
curve (dnorm (x,
              mean(datos$age),
              sd(datos$age)),
       lwd = 2,
       col = "blue",
       add = T)
legend("topleft",
       c("curva observada", "curva (normal) teórica"),
       lty = 1,
       lwd = 2,
       col = c ("black",
                "blue"),
       bty = "n",
       cex = 0.8)
qqnorm (datos$age)
qqline (datos$age)
```



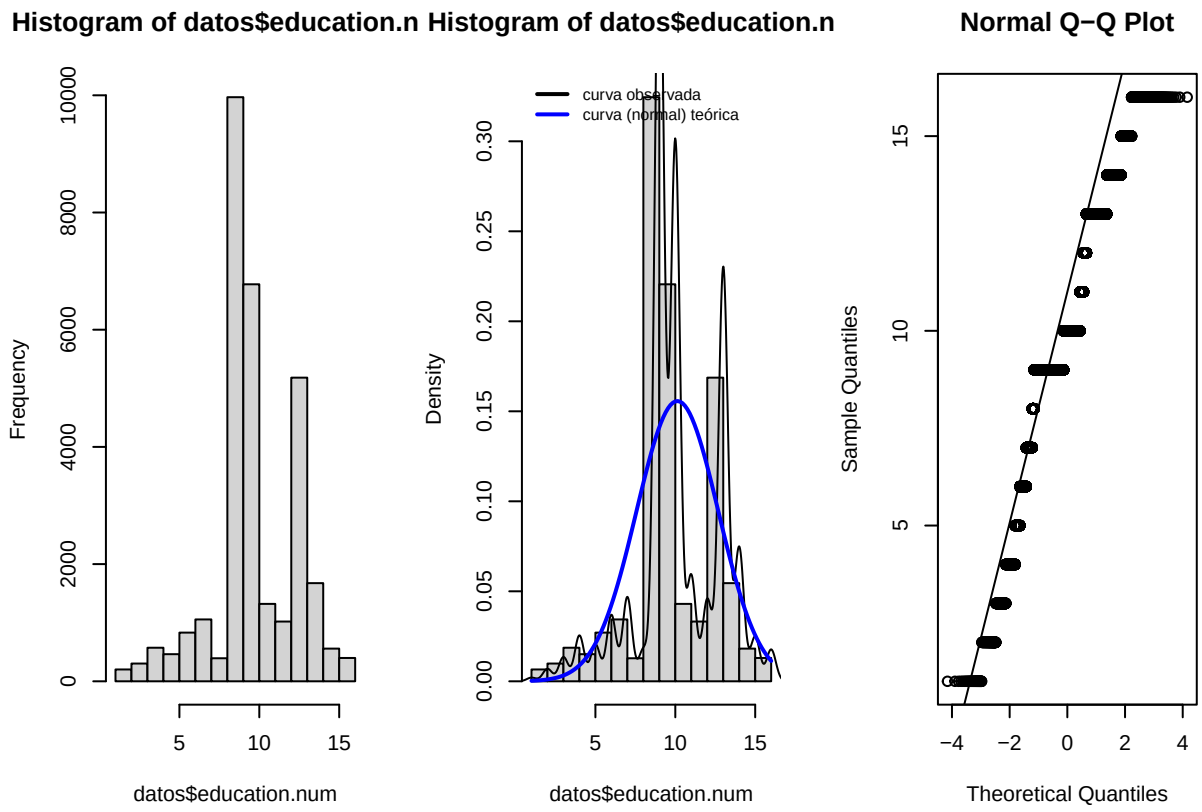

```
nortest::lillie.test (datos$education.num)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality
##  test
##
## data:  datos$education.num
## D = 0.20518, p-value < 2.2e-16
```

El valor p-value del test de Lilliefors es mucho menor al 0.05 por tanto se rechaza la hipótesis nula y se concluye que la variable número de años de estudios no sigue una distribución normal. Visualmente quedaría

```
par (mfrow = c(1,3))
hist (datos$education.num)
hist (datos$education.num,
      freq=F)
lines (density (datos$education.num))
curve (dnorm (x,
              mean(datos$education.num),
              sd(datos$education.num)),
       lwd = 2,
       col = "blue",
       add = T)
legend("topleft",
       c("curva observada", "curva (normal) teórica"),
       lty = 1,
       lwd = 2,
```

```
       col = c ("black",
                "blue"),
       bty = "n",
       cex = 0.8)
qqnorm (datos$education.num)
qqline (datos$education.num)
```



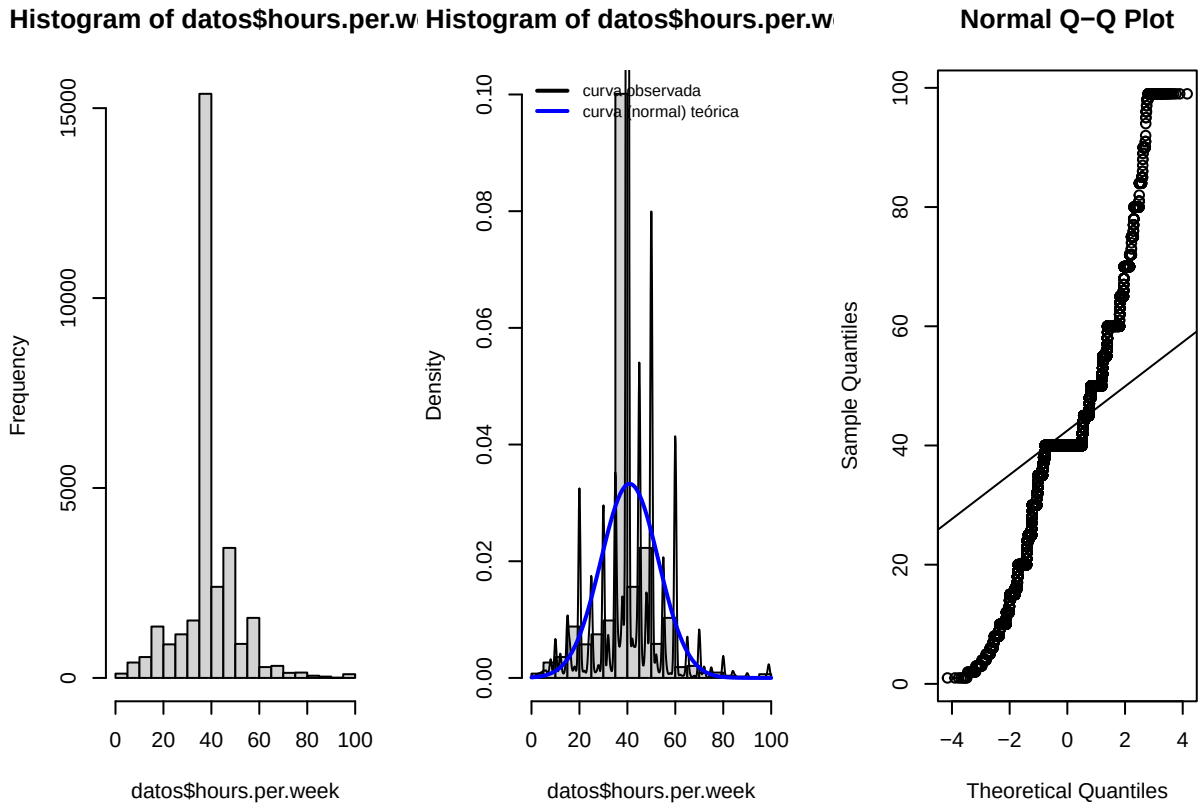

```
nortest::lillie.test (datos$hours.per.week)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality
##  test
##
## data:  datos$hours.per.week
## D = 0.24645, p-value < 2.2e-16
```

El valor p-value del test de Lilliefors es mucho menor al 0.05 por tanto se rechaza la hipótesis nula y se concluye que la variable horas de trabajo a la semana no sigue una distribución normal. Visualmente quedaría

```
par (mfrow = c(1,3))
hist (datos$hours.per.week)
hist (datos$hours.per.week,
      freq=F)
lines (density (datos$hours.per.week))
curve (dnorm (x,
              mean(datos$hours.per.week),
              sd(datos$hours.per.week)),
```

```
       lwd = 2,
       col = "blue",
       add = T)
legend("topleft",
       c("curva observada", "curva (normal) teórica"),
       lty = 1,
       lwd = 2,
       col = c ("black",
                "blue"),
       bty = "n",
       cex = 0.8)
qqnorm (datos$hours.per.week)
qqline (datos$hours.per.week)
```



Homogeneidad Implementamos el test de Fligner-Killeen, recordemos que se trata de la alternativa no paramétrica, utilizada cuando los datos no cumplen con la condición de normalidad.

La hipótesis nula asume igualdad de varianzas en los diferentes grupos de datos, por lo que p-valores inferiores al nivel de significancia indican heterocedasticidad.

En este caso, estudiaremos las diferencias en la varianza en los grupos de edad, horas semanales y estudios con la probabilidad de tener un salario superior a los 50k.

```
fligner.test (age ~ salary,
              data = datos)
```

```
##
```

```
##  Fligner-Killeen test of homogeneity of
##  variances
##
## data:  age by salary
## Fligner-Killeen:med chi-squared = 597.28, df
## = 1, p-value < 2.2e-16
```

```
fligner.test (hours.per.week ~ salary,
              data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of
##  variances
##
## data:  hours.per.week by salary
## Fligner-Killeen:med chi-squared = 34.465, df
## = 1, p-value = 4.341e-09
```

```
fligner.test (education.num ~ salary,
              data = datos)
```

```
##
##  Fligner-Killeen test of homogeneity of
##  variances
##
## data:  education.num by salary
## Fligner-Killeen:med chi-squared = 119.31, df
## = 1, p-value < 2.2e-16
```

4.3.- Aplicación de pruebas estadísticas para comparar los grupos de datos. En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

```
cor.test (datos$salary,
          datos$age,
          method = "spearman",
          exact=FALSE)
```

4.3.1 ¿Qué variables cuantitativas influyen más en el salario?

```
##
##  Spearman's rank correlation rho
##
## data:  datos$salary and datos$age
## S = 3.4916e+12, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.2772296
```

```
cor.test (datos$salary,
          datos$education.num,
          method = "spearman",
          exact=FALSE)
```

```
##
```

```
##  Spearman's rank correlation rho
##
## data:  datos$salary and datos$education.num
## S = 3.2363e+12, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.3300795
```

```
cor.test (datos$salary,
          datos$hours.per.week,
          method = "spearman",
          exact=FALSE)
```

```
##
##  Spearman's rank correlation rho
##
## data:  datos$salary and datos$hours.per.week
## S = 3.5448e+12, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.2662159
```

4.3.2 ¿La probabilidad de tener un salario superior a 50K aumenta si el individuo es un hombre de raza blanca? Se crea las muestras por sexo.

```
datos.male.salary <- datos[datos$sex == "Male",]$salary
datos.female.salary <- datos[datos$sex == "Female",]$salary
```

Mann-Whitney test.

```
wilcox.test(datos.female.salary,
            datos.male.salary,
            alternative = "less")
```

```
##
##  Wilcoxon rank sum test with continuity
##  correction
##
## data:  datos.female.salary and datos.male.salary
## W = 82539763, p-value < 2.2e-16
## alternative hypothesis: true location shift is less than 0
```

Se crea las muestras por raza.

```
datos.white.salary <- datos[datos$race == "White",]$salary
datos.black.salary <- datos[datos$race == "Black",]$salary
datos.asian.salary <- datos[datos$race == "Asian-Pac-Islander",]$salary
datos.indian.salary <- datos[datos$race == "Amer-Indian-Eskimo",]$salary
```

Mann-Whitney test.

```
wilcox.test(datos.black.salary,
            datos.white.salary,
            alternative = "less")
```

```
##
```

```
##  Wilcoxon rank sum test with continuity
##  correction
##
## data:  datos.black.salary and datos.white.salary
## W = 33125646, p-value < 2.2e-16
## alternative hypothesis: true location shift is less than 0
```

```
wilcox.test(datos.asian.salary,
            datos.white.salary,
            alternative = "less")
```

```
##
##  Wilcoxon rank sum test with continuity
##  correction
##
## data:  datos.asian.salary and datos.white.salary
## W = 12990645, p-value = 0.8383
## alternative hypothesis: true location shift is less than 0
```

```
wilcox.test(datos.indian.salary,
            datos.white.salary,
            alternative = "less")
```

```
##
##  Wilcoxon rank sum test with continuity
##  correction
##
## data:  datos.indian.salary and datos.white.salary
## W = 3215168, p-value = 1.439e-08
## alternative hypothesis: true location shift is less than 0
```

```
# Regresores cuantitativos
educationNum = datos$education.num
ageIndividual = datos$age
hoursPerWeek = datos$hours.per.week
# Regresores cualitativos
sexIndividual = datos$sex
raceIndividual = datos$race
workclassInd = datos$workclass
# Variable a predecir
salary50k = datos$salary
# Generación de varios modelos
# No age, sex or race
modelo1 <- lm(salary50k ~ educationNum +
                hoursPerWeek +
                workclassInd,
              data = datos)
# No sex or race
modelo2 <- lm(salary50k ~ educationNum +
                hoursPerWeek +
                workclassInd +
                ageIndividual,
              data = datos)
# No educationNum
```

```r
modelo3 <- lm(salary50k ~ ageIndividual +
                sexIndividual +
                raceIndividual +
                workclassInd +
                hoursPerWeek,
              data = datos)
# No educationNum or workclass
modelo4 <- lm(salary50k ~ sexIndividual +
                raceIndividual +
                hoursPerWeek,
              data = datos)
# No education or age
modelo5 <- lm(salary50k ~ sexIndividual +
                raceIndividual +
                workclassInd +
                hoursPerWeek,
              data = datos)
# Only education, hours and workclass
modelo6 <- lm(salary50k ~ educationNum +
                hoursPerWeek +
                workclassInd,
              data = datos)
# Only sex, race and education
modelo7 <- lm(salary50k ~ educationNum +
                sexIndividual +
                raceIndividual,
              data = datos)
# Only education
modelo8 <- lm(salary50k ~ educationNum,
              data = datos)
```

```r
# Tabla con los coeficientes de determinación de cada modelo
tabla.coeficientes <- matrix(c(1,
                               summary(modelo1)$r.squared,
                               2,
                               summary(modelo2)$r.squared,
                               3,
                               summary(modelo3)$r.squared,
                               4,
                               summary(modelo4)$r.squared,
                               5,
                               summary(modelo5)$r.squared,
                               6,
                               summary(modelo6)$r.squared,
                               7,
                               summary(modelo7)$r.squared,
                               8,
                               summary(modelo8)$r.squared),
                             ncol = 2,
                             byrow = TRUE)

colnames (tabla.coeficientes) <- c("Modelo",
                                   "R^2")
```

```
tabla.coeficientes
```

4.3.3. Modelo de regresión lineal

```
##      Modelo        R^2
## [1,]      1 0.14961246
## [2,]      2 0.19086448
## [3,]      3 0.13359781
## [4,]      4 0.08592147
## [5,]      5 0.09519242
## [6,]      6 0.14961246
## [7,]      7 0.16021941
## [8,]      8 0.11198415
```

```
train <- datos
```

```
train <- dplyr::select (train,
                        -id,
                        -age_cat,
                        -fnlwgt,
                        -education,
                        -native.country)
```

4.3.4. Modelo de regresión logística

Cambiamos el tipo de datos a caracter.

```
train$salary <- as.character (train$salary)
class (train$salary)
```

```
## [1] "character"
```

```
set.seed (1000)

trainCtrl = trainControl (method = "cv",
                          number = 10)

regresionModelo = train (salary ~ age +
                           workclass +
                           education.num +
                           marital.status +
                           occupation +
                           relationship +
                           race +
                           sex +
                           hours.per.week,
                         trControl = trainCtrl,
                         method = "gbm",
                         data = train,
                         verbose = FALSE)
```

Confusion Matrix of Training data

```
confusionMatrix (factor(train$salary),
                 predict (regresionModelo,
                          train))
```

```
## Confusion Matrix and Statistics
```

```
##
##           Reference
## Prediction     0     1
##          0 21308  1760
##          1  3271  4379
##
##                Accuracy : 0.8362
##                  95% CI : (0.832, 0.8403)
##     No Information Rate : 0.8001
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5312
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8669
##             Specificity : 0.7133
##          Pos Pred Value : 0.9237
##          Neg Pred Value : 0.5724
##              Prevalence : 0.8001
##          Detection Rate : 0.6937
##    Detection Prevalence : 0.7510
##       Balanced Accuracy : 0.7901
##
##        'Positive' Class : 0
##
```

Elaboramos un nuevo modelo.

```
set.seed (1001)
trainCtrl = trainControl (method = "cv",
                          number = 10)


regresionModelo2 = train (salary ~ age +
                            workclass +
                            education.num +
                            marital.status +
                            occupation +
                            relationship +
                            hours.per.week,
                          trControl = trainCtrl,
                          method = "gbm",
                          data = train,
                          verbose = FALSE)
```

Confusion Matrix of Training data

```
confusionMatrix (factor(train$salary),
                 predict (regresionModelo2,
                          train))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
```

```
##          0 21348  1720
##          1  3312  4338
##
##                Accuracy : 0.8362
##                  95% CI : (0.832, 0.8403)
##     No Information Rate : 0.8028
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5293
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8657
##             Specificity : 0.7161
##          Pos Pred Value : 0.9254
##          Neg Pred Value : 0.5671
##              Prevalence : 0.8028
##          Detection Rate : 0.6950
##    Detection Prevalence : 0.7510
##       Balanced Accuracy : 0.7909
##
##        'Positive' Class : 0
##
```

5.- Representación de los resultados a partir de tablas y gráficas.

Vamos a estudiar la correlación entre los valores del dataset. Como la función de correlación solo funciona con valores numéricos, hemos tenido que hacer un pequeño ajuste, creando un nuevo dataset.

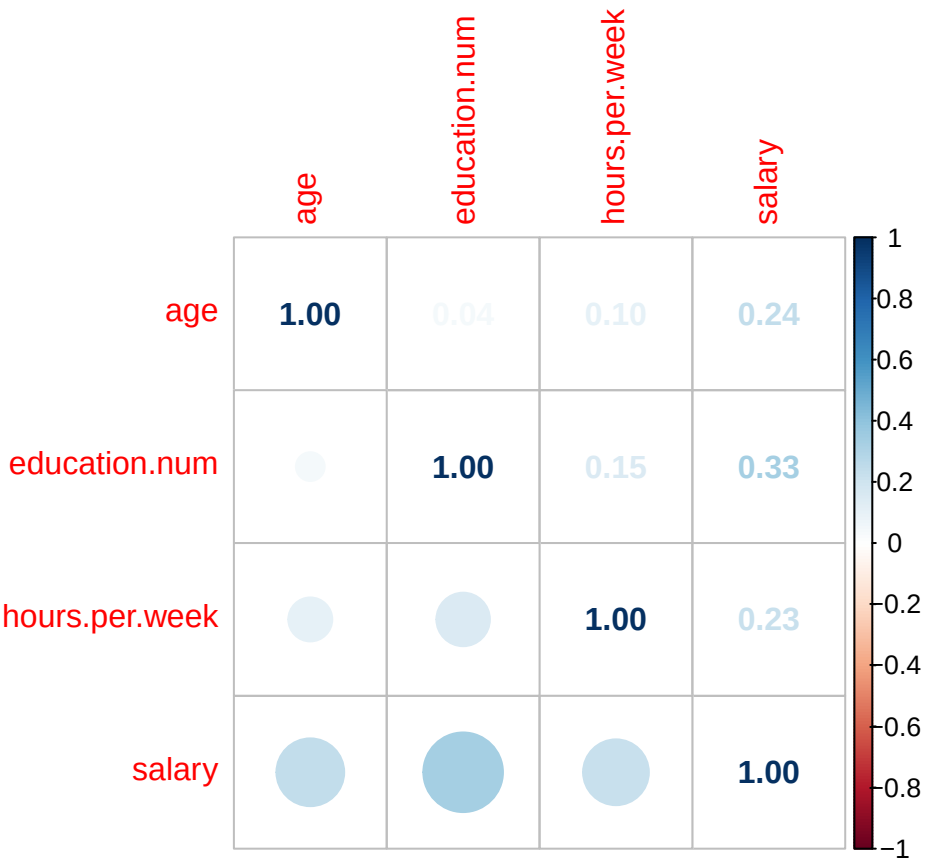
```
datosSalary <- datos[ , c(2,6,12,14)]
```

Convertimos los valores a numéricos.

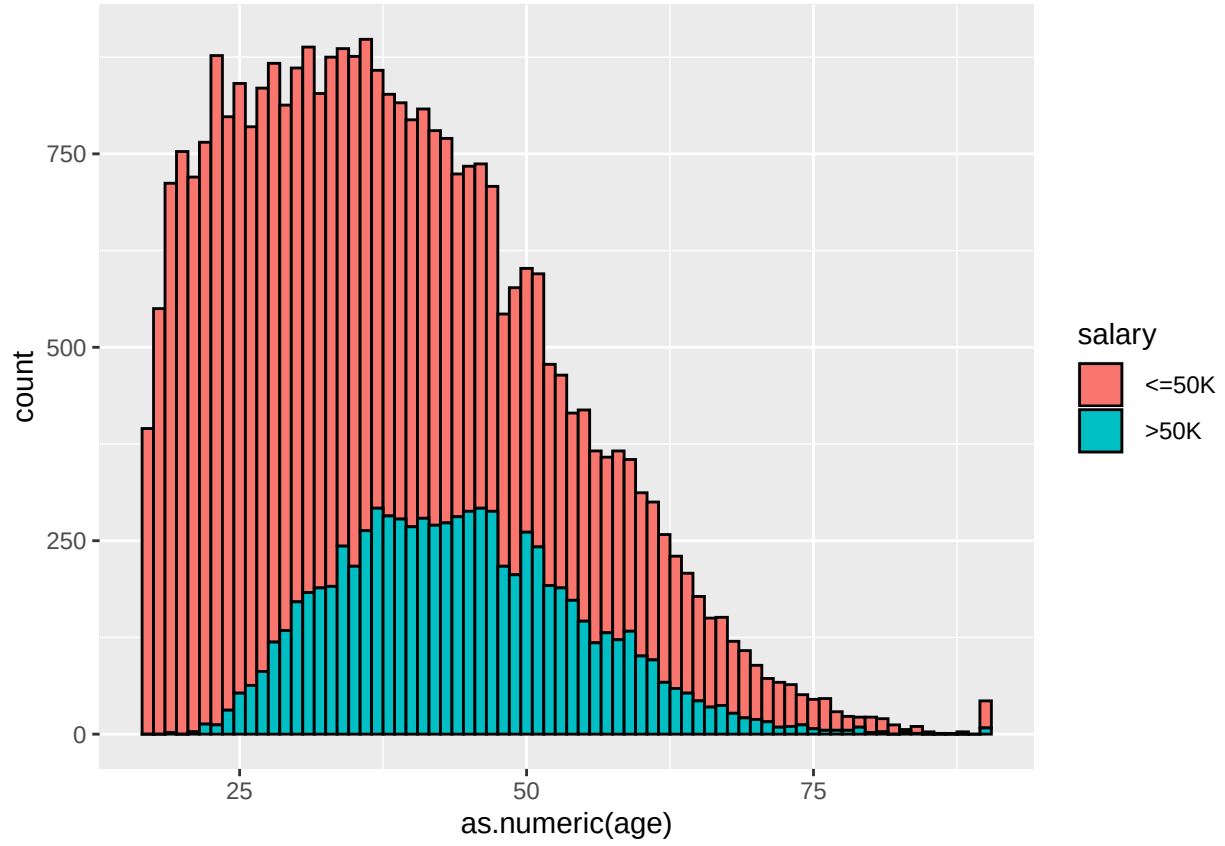
```
datosSalary[] <- lapply (datosSalary,
                         as.numeric)
```

Creamos el gráfico de correlaciones.

```
cor_matrix <- cor(datosSalary,
                  use = 'complete.obs')
corrplot.mixed (cor_matrix,
                lower = "circle",
                upper = "number",
                tl.pos = "lt",
                diag = "u")
```

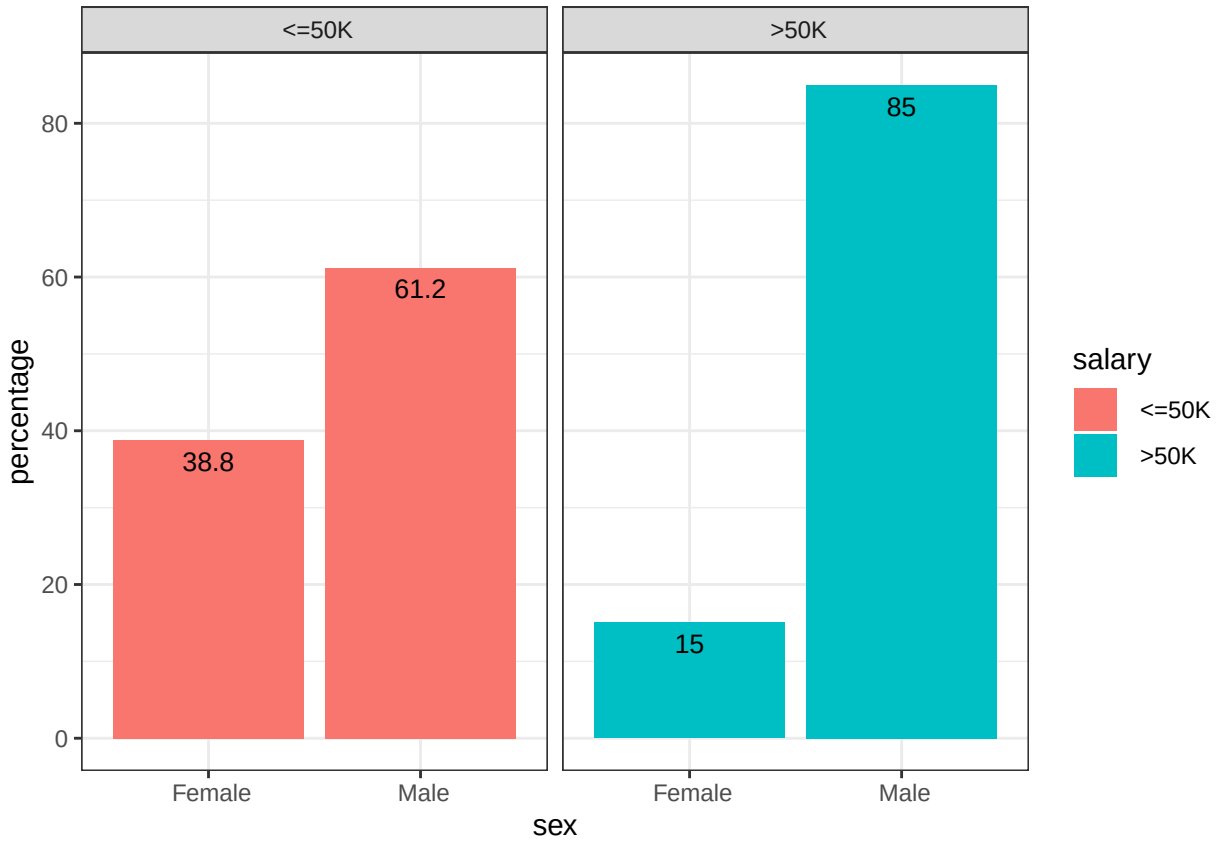



```
# edad-income
ggplot (unmodifiedDatos) +
  aes(x = as.numeric(age),
      group = salary,
      fill = salary) +
  geom_histogram(binwidth=1,
                 color='black')
```

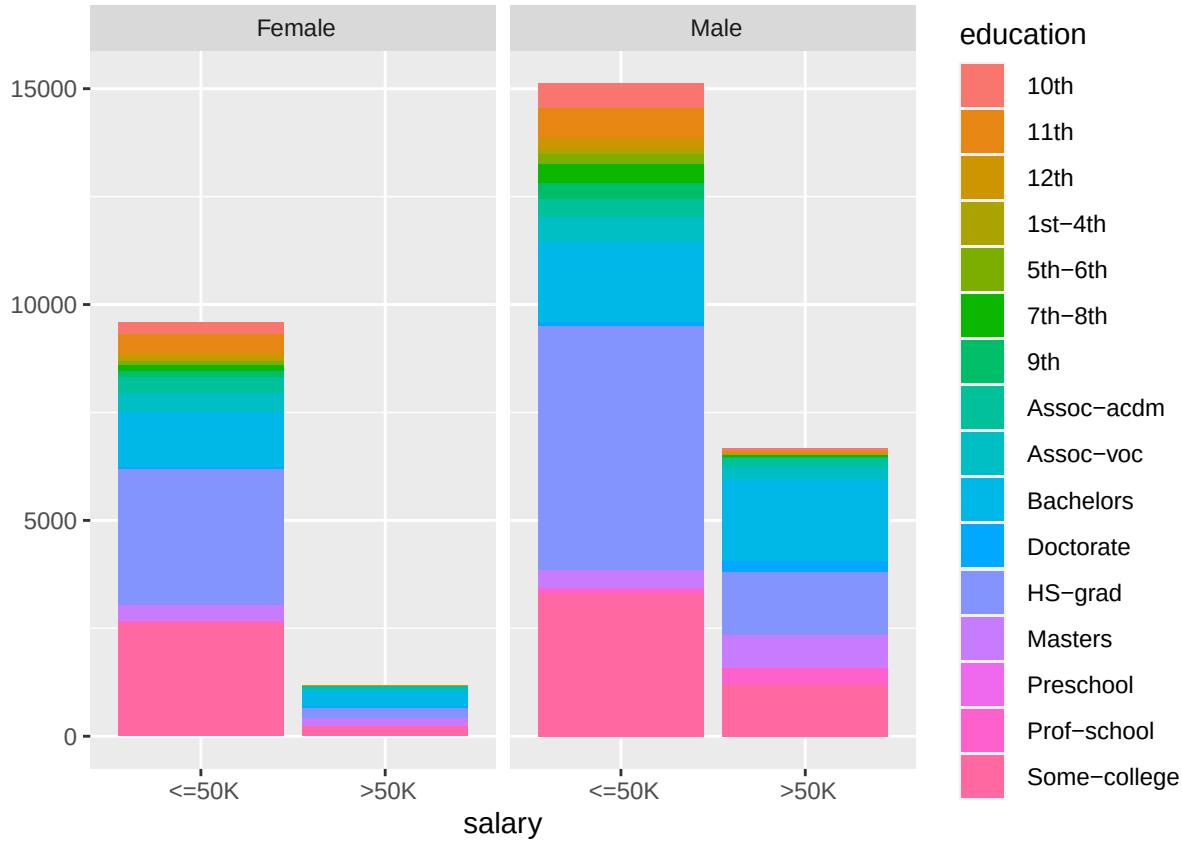



```
# Sex (percentage) - salary
visualizacion_Genero <- unmodifiedDatos %>%
  group_by (salary,
            sex) %>%
  tally() %>%
  complete (sex,
            fill = list(n = 0)) %>%
  mutate (percentage = n / sum(n) * 100)

ggplot(visualizacion_Genero, aes(sex, percentage, fill = salary)) +
  geom_bar(stat = 'identity', position=position_dodge()) +
  geom_text(aes(label=round(percentage, digits = 1)), vjust=1.6,
            color="black",position = position_dodge(0.9), size=3.5) +
  facet_wrap(~ salary)+
  theme_bw()
```

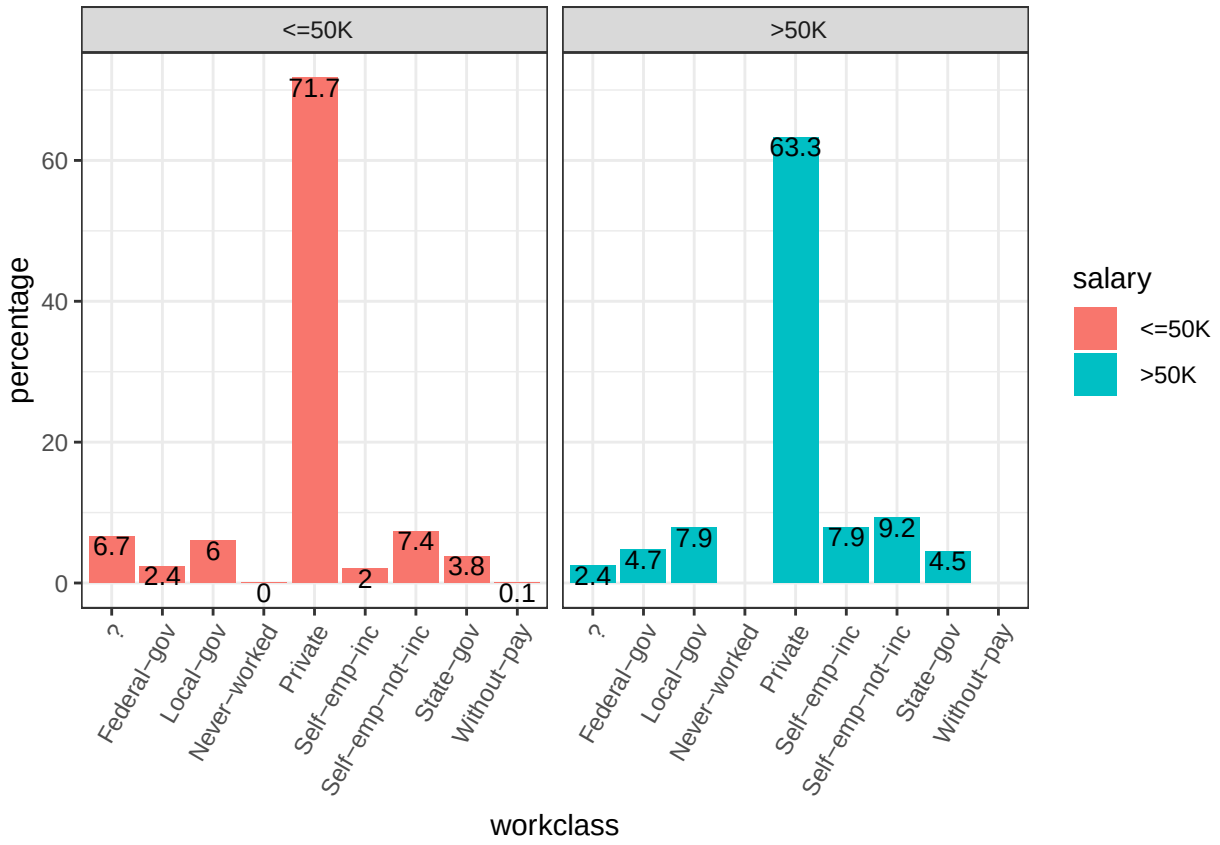



```
# Education (percentage) - salary
qplot(salary,
      data = unmodifiedDatos,
      fill = education) +
  facet_grid (.~sex)
```



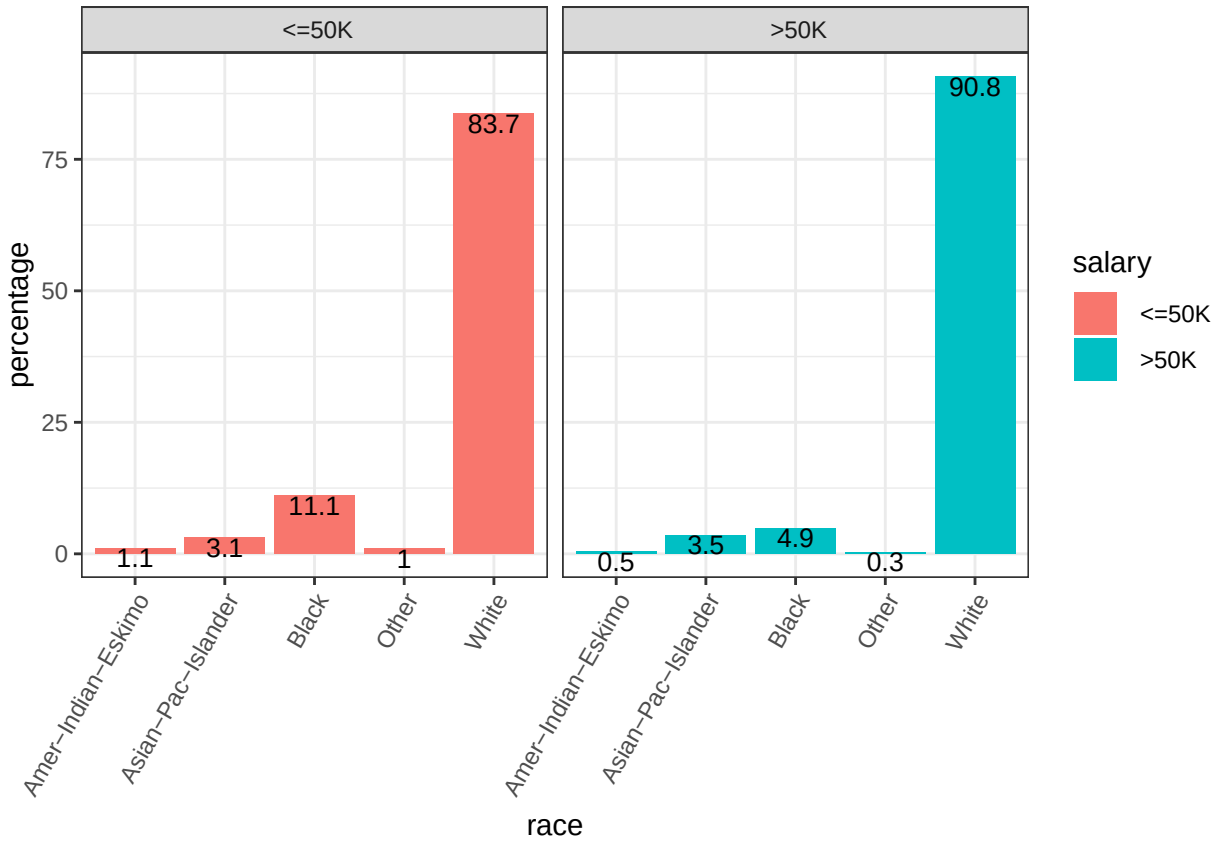

```
# Workclass (percentage) - salary
visualizacion_Genero <- unmodifiedDatos %>%
  group_by (salary, workclass) %>%
  tally () %>%
  complete (workclass,
            fill = list(n = 0)) %>%
  mutate (percentage = n / sum(n) * 100)

ggplot (visualizacion_Genero,
        aes (workclass,
             percentage,
             fill = salary)) +
  geom_bar (stat = 'identity',
            position = position_dodge()) +
  geom_text (aes (label = round(percentage,
                                digits = 1)),
             vjust = 1,
             color = "black",
             position = position_dodge (0.9),
             size = 3.5) +
  facet_wrap (~ salary) +
  theme_bw () +
  theme (axis.text.x=element_text (angle=60,
                                   hjust=1))
```




```
# Race (percentage) - salary
visualizacion_Genero <- unmodifiedDatos %>%
  group_by (salary,
            race) %>%
  tally () %>%
  complete (race,
            fill = list (n = 0)) %>%
  mutate (percentage = n / sum(n) * 100)

ggplot (visualizacion_Genero,
        aes (race,
             percentage,
             fill = salary)) +
  geom_bar (stat = 'identity',
            position = position_dodge ()) +
  geom_text (aes (label = round (percentage,
                                 digits = 1)),
             vjust = 1,
             color = "black",
             position = position_dodge(0.9),
             size=3.5) +
  facet_wrap (~ salary) +
  theme_bw () +
  theme (axis.text.x = element_text (angle=60,
                                     hjust=1))
```



6. Resolución del problema. A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

Por la distribución de la variable salario, es evidente que existe un sesgo o discriminación hacia un tipo determinado de individuos.

Entre los 30 y 50 años, son cuando se alcanza la mayor probabilidad de obtener un mayor salario.

Los hombres, estadísticamente, son más propensos a ganar más de 50K. También, son más propensos a ganar menos de 50K, pero esto solamente evidenciaría que hay más población de hombres trabajadores que de mujeres.

La educación, aunque es la variable más influyente, mayores niveles de educación no parece aumentar la probabilidad de ganar más. El grupo más grande estaría conformado por aquellos con “Bachelors”, tanto para ambos sexos.

Dentro del sector laboral, el sector privado es donde la probabilidad de ganar más es mayor.

La población de raza blanca tiene mayor acceso a la vida laboral, y de igual forma, su salario es potencialmente mayor. También, se observa una discriminación evidente en el grupo de raza “negra” y “india”, los cuales son tienen menor probabilidad de ganar más de 50K.

Si atendemos a los diferentes tests estadísticos, extraemos las siguientes conclusiones:

Entre las diferentes variables cuantitativas analizadas (education, age, hours.per.week), vemos que la más condicionante es la variable education. Por tanto, concluimos que esta variable tiene mayor peso que las otras dos en determinar el salario. Aunque las diferencias son mínimas, la escala de influencia sería la siguiente: education » age > hours.per.week.

En el contraste de hipótesis, hemos determinado sesgos. Los hombres tienen más probabilidad de ganar más

que las mujeres. Los individuos de raza blanca son más propensos a ganar más que los de otras razas (a excepción de los de raza asiática).

Los modelos de regresión lineal no nos servirían para predecir el salario, pero nos dan una idea de la influencia de cada variable. La variable education sigue siendo la más influente, las variables race o sex aunque influyen no condicionan en gran medida el modelo.

En el modelo de regresión logística, hemos podido observar un patrón similar. Aunque las variables race o sex, se ha determinado como influyentes y se ha concluido la existencia de un sesgo, no incluir estas variables en el modelo apenas ha condicionado la precisión, sensibilidad o especificidad de este. No podríamos concluir que dichas variables no influyen, porque estaríamos cayendo en error, sino que deberíamos concluir que los hombres de raza blanca, conforman la mayoría de los encuestados y por tanto el modelo es preciso para este colectivo, pero de ser aplicado a los grupos discriminados, obtendríamos un modelo totalmente erróneo.

7. Código: Hay que adjuntar el código, preferiblemente en R, con el que se ha realizado la limpieza, análisis y representación de los datos. Si lo preferís, también podéis trabajar en Python.

El código del proyecto así como todo lo necesario para su ejecución puede encontrarse en

https://github.com/UOCPgarcia/Factores_vs_Salario_Cleaning