A1 - Preproceso de datos

Solución

Semestre 2022.1

Índice

Introducción

El conjunto de datos está en el archivo `gpa_row.csv`, contiene la nota media de estudiantes universitarios después del primer semestre de clases (GPA: grade point average, en inglés), así como información sobre la nota de acceso, la cohorte de graduación en el instituto y algunas características de los estudiantes.

Este conjunto de datos surge de una encuesta realizada a una muestra representativa de estudiantes de una universidad de EEUU (por razones de confidencialidad el conjunto de datos no incluye el nombre de la universidad). Las variables incluidas en el conjunto de datos son:

- sat: nota de acceso (medida en escala de 400 a 1600 puntos)
- tothrs: horas totales cursadas en el semestre
- hsize: numero total de estudiantes en la cohorte de graduados del bachillerato (en cientos)
- hsrank: ranking del estudiante, dado por la nota media del bachillerato, en su cohorte de graduados del bachillerato
- hsperc: ranking relativo del estudiante (hsrank/hsize).
- colgpa: nota media del estudiante al final del primer semestre (medida en escala de 0 a 4 puntos)
- athlete: indicador de si el estudiante practica algún deporte en la universidad
- female: indicador de si el estudiante es mujer
- white: indicador de si el estudiante es de raza blanca o no
- black: indicador de si el estudiante es de raza negra o no

El objetivo de esta actividad es preparar el archivo para su posterior análisis. Para ello, se examinará el archivo para detectar y corregir posibles errores, inconsistencias y valores perdidos. Además se presentará una breve estadística descriptiva con gráficos.

Por otra parte, se realizará un informe ejecutivo que resumirá lo realizado. Constará de dos partes:

1. Documentar todos los cambios realizados en los datos originales.
2. Breve resumen de las características más destacables de cada variable.

Criterios de verificación y de normalización de las variables:

A continuación se muestran los criterios con los que deben limpiarse los datos del conjunto:

1. Verificar que las variables de tipo indicador deben tener sólo el valor TRUE o FALSE (mayúsculas y sin espacios en blanco) y deben codificarse como variables categóricas ("factor"). En caso de que no se cumpla, es necesario corregirlo.
2. En los datos de naturaleza numéricas, el símbolo de separador decimal es el punto y no la coma. Además, si se presenta la unidad de la variable es necesario eliminarla para convertir la variable a tipo numérico.
3. Comprobar si se cumple el rango de valores posibles en las variables donde se tiene esta información:
   - ‘sat‘ : escala de 400 a 1600 puntos
   - ‘colgpa‘ : escala de 0 a 4 puntos
4. Revisar si los valores de la variable ‘hsperc‘ se ha calculado correctamente a partir de ‘hsrank / hsize‘ con tres decimales de precisión. En caso contrario, modificarlo.

Nota importante a tener en cuenta para entregar la actividad:

- Es necesario entregar el archivo Rmd y el archivo de salida (PDF o html). El archivo de salida debe incluir: el código y el resultado de la ejecución del código (paso a paso).
- Se respetará la misma numeración de los apartados que el enunciado.

- No se pueden realizar listas completas del conjunto de datos en la solución. Esto generaría un documento con cientos de páginas y dificulta la revisión del texto. Para comprobar las funcionalidades del código sobre los datos, se pueden utilizar las funciones **head** y **tail** que sólo muestran unas líneas del archivo de datos.
- Se valora la precisión de los términos utilizados (es necesario utilizar de forma precisa la terminología de la estadística).
- Se valora también la concisión en la respuesta. No se trata de realizar explicaciones muy largas o documentos muy extensos. Hay que explicar el resultado y argumentar la respuesta a partir de los resultados obtenidos de forma clara y concisa.

Para realizar el preproceso del fichero, seguir los pasos que se indican a continuación.

---

1 Carga del archivo

Cargue el archivo de datos y examine el tipo de datos con el que R ha interpretado cada variable.

Indique qué variables son de naturaleza numérica, aunque R lo haya podido interpretar de forma distinta. En caso de que el tipo de variable que ha otorgado R no coincida con el tipo que le correspondería, deberá aplicar la transformación correspondiente cuando realice la normalización de la variable (apartado siguiente).

```
#FUNCIÓN PARA DOCUMENTAR LOS CAMBIOS INTRODUCIDOS EN EL PREPROCESAMIENTO
report <- function( ds, row="", message=""){
       i <- nrow(ds)-1
       rw <- data.frame(id=i+1, row, message)
       ds <- rbind( ds, rw )

     return (ds)
}
```

```
ds<-read.csv("gpa_row.csv",stringsAsFactors=TRUE)

# We get the dimensions of the dataset, structure and content
dim(ds)
```

```
## [1] 4137   10
```

```
str(ds)
```

```
## 'data.frame':    4137 obs. of  10 variables:
##  $ sat    : int  920 1170 810 940 1180 980 880 980 1240 1230 ...
##  $ tothrs : Factor w/ 125 levels "100h","101h",..: 67 46 42 64 46 16 103 79 46 45 ...
##  $ hsize  : Factor w/ 649 levels "0,30000001","0,40000001",..: 9 649 122 553 223 277 324 277 379 9 .
##  $ hsrank : int  4 191 42 252 86 41 161 101 161 3 ...
##  $ hsperc : num  40 20.3 35.3 44.1 40.2 ...
##  $ colgpa : num  2.04 4 1.78 2.42 2.61 ...
##  $ athlete: Factor w/ 4 levels "false","FALSE",..: 4 2 4 2 2 2 2 2 2 2 ...
##  $ female : logi  TRUE FALSE FALSE FALSE FALSE TRUE ...
##  $ white  : Factor w/ 6 levels "  TRUE","false",..: 3 5 5 5 5 5 3 5 5 5 ...
##  $ black  : Factor w/ 6 levels "  FALSE","false",..: 3 3 3 3 3 3 3 3 3 3 ...
```

```
head(ds)
```

```
##    sat tothrs     hsize hsrank   hsperc colgpa athlete female white black
## 1  920    43h       0.1      4 40.00000   2.04    TRUE   TRUE FALSE FALSE
## 2 1170    18h 9.3999996    191 20.31915   4.00   FALSE  FALSE  TRUE FALSE
```

```
## 3  810    14h 1.1900001     42 35.29412   1.78    TRUE  FALSE  TRUE FALSE
## 4  940    40h      5.71    252 44.13310   2.42   FALSE  FALSE  TRUE FALSE
## 5 1180    18h 2.1400001     86 40.18692   2.61   FALSE  FALSE  TRUE FALSE
## 6  980   114h 2.6800001     41 15.29851   3.03   FALSE   TRUE  TRUE FALSE
```

```
summary(ds)
```

```
##       sat           tothrs           hsize          hsrank
##  Min.   : 470   17h    : 305   0.1      : 115   Min.   :  1.00
##  1st Qu.: 940   16h    : 279   2.3399999:  49   1st Qu.: 11.00
##  Median :1030   15h    : 226   2.8      :  49   Median : 30.00
##  Mean   :1030   14h    : 167   2.1099999:  41   Mean   : 52.83
##  3rd Qu.:1120   18h    : 153   2.03     :  37   3rd Qu.: 70.00
##  Max.   :1540   13h    : 146   2.3800001:  36   Max.   :634.00
##                 (Other):2861   (Other)  :3810
##      hsperc            colgpa       athlete       female            white
##  Min.   : 0.1667   Min.   :0.000   false:  11   Mode :logical     TRUE :   2
##  1st Qu.: 6.4328   1st Qu.:2.210   FALSE:3932   FALSE:2277      false  :   3
##  Median :14.5963   Median :2.660   true :   1   TRUE :1860      FALSE  : 305
##  Mean   :19.2406   Mean   :2.655   TRUE : 193                   true   :   9
##  3rd Qu.:27.7108   3rd Qu.:3.120                                TRUE   :3814
##  Max.   :92.0000   Max.   :4.000                                TRUE   :   4
##                    NA's   :41
##       black
##    FALSE :   3
##  false   :  10
##  FALSE   :3890
##  FALSE   :   5
##  TRUE    : 228
##  TRUE    :   1
##
```

```
id.factor   <- c(7:10)
id.num      <- c(1:6)
var.factor  <- colnames(ds)[id.factor]
var.num     <- colnames(ds)[id.num]

info <- data.frame(id=1, row="",
                   message= paste0("n. row = ", nrow(ds),"; ",
                                   "n. col= ", ncol(ds), "; ",
                                   "n. var num. = ",  length(id.num),"; ",
                                   "n. var cualit. = ", length(id.factor)))
```

Haurien de ser variables qualitatives (factor): **athlete, female, white, black**

Haurien de ser variables quantitatives (numèriques): **sat, tothrs, hsize, hsrank, hsperc, colgpa**

2 Normalización de las variables cualitativas

2.1 Athlete

Normalizar la variable `Athlete` según las indicaciones proporcionadas.

```
  # Revisión variable
  table( ds$athlete )
```

```
##
```

```
## false FALSE  true  TRUE
##    11  3932     1   193
```

```
  #Report of changes
  idx <- which( ds$athlete =="false")
  idx
```

```
##  [1]   65  385  720  939 1293 1312 2330 2400 2933 3193 3543
```

```
  info <- report(info, row=paste(idx,collapse=", "), "athlete: false  -> FALSE")

  idx <- which( ds$athlete =="true")
  idx
```

```
## [1] 876
```

```
  info <- report(info, row=paste(idx,collapse=", "), "athlete: true  -> TRUE")

  #---------------------------------
  # Change to capital letters
  ds$athlete <- str_to_upper(ds$athlete)

  # Change to factor
  ds$athlete <- factor(ds$athlete)

  # checking
  table( ds$athlete )
```

```
##
## FALSE  TRUE
##  3943   194
```

2.2 Female

Normalizar la variable `Female` según las indicaciones proporcionadas.

```
  # Revisión variable
  table( ds$female )
```

```
##
## FALSE  TRUE
##  2277  1860
```

```
  # change to factor
  ds$female <- factor(ds$female)


  # Tot correcte
```

2.3 Black

Normalizar la variable `Black` según las indicaciones proporcionadas.

```
  # Revisión variable
  table( ds$black )
```

```
##
##    FALSE    false    FALSE FALSE        TRUE  TRUE
##        3       10     3890        5      228        1
```

```
  #Report of changes
  idx <- grep('[[:space:]]', ds$black)
  length(idx)
```

```
## [1] 9
```

```
  info <- report(info, row=paste(idx,collapse=", "), "black: Eliminado espacios en blanco")

  idx <- which( ds$black =="false")
  idx
```

```
##  [1] 1255 1785 2227 2424 2450 2913 3076 3102 3803 3868
```

```
  info <- report(info, row=paste(idx,collapse=", "), "black: false  -> FALSE")

  #---------------------------------
  # Remove blank spaces
  ds$black <- str_trim(ds$black)
  table( ds$black )
```

```
##
## false FALSE  TRUE
##    10  3898   229
```

```
  # Change to capital letters
  ds$black <- str_to_upper(ds$black)

  # Change to factor
  ds$black <- factor(ds$black)

  # checking
  table( ds$black )
```

```
##
## FALSE  TRUE
##  3908   229
```

2.4 White

Normalizar la variable `white` según las indicaciones proporcionadas.

```
  # Revisión variable
  table( ds$white )
```

```
##
##    TRUE   false   FALSE    true    TRUE TRUE
##       2       3     305       9    3814       4
```

```
  #Report of changes
  idx <-grep('[[:space:]]', ds$white)
  length(idx)
```

```
## [1] 6
```

```
  info <- report(info, row=paste(idx,collapse=", "), "white: Eliminar espacios en blanco")

  idx <- which( ds$white =="false")
  idx
```

```
## [1] 1929 2922 3536
```

```
  info <- report(info, row=paste(idx,collapse=", "), "white: false  -> FALSE")

  idx <- which( ds$white =="true")
  idx
```

```
## [1]  461  922 1007 1947 2810 2969 3129 4029 4030
```

```
  info <- report(info, row=paste(idx,collapse=", "), "white: true  -> TRUE")

  #----------------------------------
  # Remove  blank spaces
  ds$white <- str_trim(ds$white)
  table( ds$white )
```

```
##
## false FALSE  true  TRUE
##     3   305     9  3820
```

```
  # Change to capital letters
  ds$white <- str_to_upper(ds$white)

  # Change to factor
  ds$white <- factor(ds$white)

  # checking
  table( ds$white )
```

```
##
## FALSE  TRUE
##   308  3829
```

3 Normalización de las variables cuantitativas

Inspeccionar los valores de los datos cuantitativos y realizar las normalizaciones oportunas siguiendo los criterios especificados anteriormente. Estas normalizaciones tienen como objetivo uniformizar los formatos. Si hay valores perdidos o valores extremos, se tratarán más adelante.

Al realizar estas normalizaciones, se debe demostrar que la normalización sobre cada variable ha dado el resultado esperado. Por lo tanto, se recomienda mostrar un fragmento del archivo de datos resultante. Para evitar mostrar todo el conjunto de datos, se puede mostrar una parte del mismo, con las funciones **head** y/o **tail**.

Seguid el orden de los apartados.

3.1 Nota de acceso

Revise el formato de la variable `sat` y realice las revisiones o transformaciones oportunas según los criterios especificados anteriormente.

```
head(ds$sat,8)
```

```
## [1]  920 1170  810  940 1180  980  880  980
```

```
# checking

idx <- which(ds$sat <  400 | ds$sat >  1600)
```

```
#
# All values are correct


summary(ds$sat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     470     940    1030    1030    1120    1540
```

3.2 Horas totales cursadas al semestre

Revise el formato de la variable `tothrs` y realice las transformaciones oportunas según los criterios especificados anteriormente. Si existen valores atípicos, se tratarán más adelante.

```
head(ds$tothrs,8)
```

```
## [1] 43h  18h  14h  40h  18h  114h 78h  55h
## 125 Levels: 100h 101h 102h 103h 104h 105h 106h 107h 108h 109h 10h 110h ... 9h
```

```
ds$tothrs <- as.numeric( trimws( sub('h', "", ds$tothrs ) ) )
head(ds$tothrs,8)
```

```
## [1]  43  18  14  40  18 114  78  55
```

```
# checking
class(ds$tothrs)
```

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

```
summary(ds$tothrs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    6.00   17.00   47.00   52.83   80.00  137.00
```

```
#Report of changes
 info <- report(info, row="*", "tothrs: Eliminado el texto h y seha convertido a variable numérica")
```

3.3 Nota media del estudiante al final del primer semestre

Revise el formato de la variable `colgpa` y realice las revisiones o transformaciones oportunas según los criterios especificados anteriormente.

```
head(ds$colgpa,8)
```

```
## [1] 2.04 4.00 1.78 2.42 2.61 3.03 1.84 3.05
```

```
# Ckecking

idx <- which(ds$colgpa <  0 | ds$colgpa >  4)
#
# All values are correct


summary(ds$colgpa)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   2.210   2.660   2.655   3.120   4.000      41
```

3.4 Número total de estudiants en la cohorte de graduados del bachillerato

Revise el formato de la variable **hsize** y realice las revisiones o transformaciones oportunas según los criterios especificados anteriormente.

```
class(ds$hsize)
```

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

```
head( ds$hsize, 30)
```

```
##  [1] 0.1        9.3999996  1.1900001  5.71       2.1400001  2.6800001
##  [7] 3.1099999  2.6800001  3.6700001  0.1        3.3399999  3.5899999
## [13] 3.1800001  1.92       3.6900001  2.6600001  1.45       1.76
## [19] 3.8599999  3.8299999  1.0700001  2.1700001  2.3399999  4.6300001
## [25] 5.9499998  0.91000003 4.3600001  8.1199999  0.60000002 3.76
## 649 Levels: 0,30000001 0,40000001 0,73000002 0.029999999 ... 9.3999996
```

```
ds$hsize <- as.character( ds$hsize )

##Report of changes
idx <- grep("\\,",ds$hsize)
#ds$hsize[idx]
info <- report(info, row=paste(idx,collapse=", "), "corregimos la coma por el punto decimal")

#---------------------------------
#corregimos la coma por el punto decimal
ds$hsize <- gsub("\\,","\\.", ds$hsize)
ds$hsize <- as.numeric( ds$hsize )

#ckecking
head( ds$hsize, 30)
```

```
##  [1] 0.10 9.40 1.19 5.71 2.14 2.68 3.11 2.68 3.67 0.10 3.34 3.59 3.18 1.92 3.69
## [16] 2.66 1.45 1.76 3.86 3.83 1.07 2.17 2.34 4.63 5.95 0.91 4.36 8.12 0.60 3.76
```

3.5 Ranking relativo del estudiante

Revise si la variable **hsperc** se ha obtenido correctamente según los criterios especificados anteriormente.

```
idx <- which(round((ds$hsrank/ ds$hsize),3)
               !=  round(ds$hsperc,3))
idx
```

```
##  [1]  188  201  313  657  876 2489 3438 3441 3445 3537 3753 4091
```

```
##Report of changes
info <- report(info, row=paste(idx,collapse=", "), "hsperc: recalcular los valores de hsrank/hsize")


#---------------------------------
ds$hsperc[idx] <- round(ds$hsrank[idx]/ ds$hsize[idx],3)

# Checking
idx <- which(round((ds$hsrank/ ds$hsize),3)
               !=  round(ds$hsperc,3))
idx
```

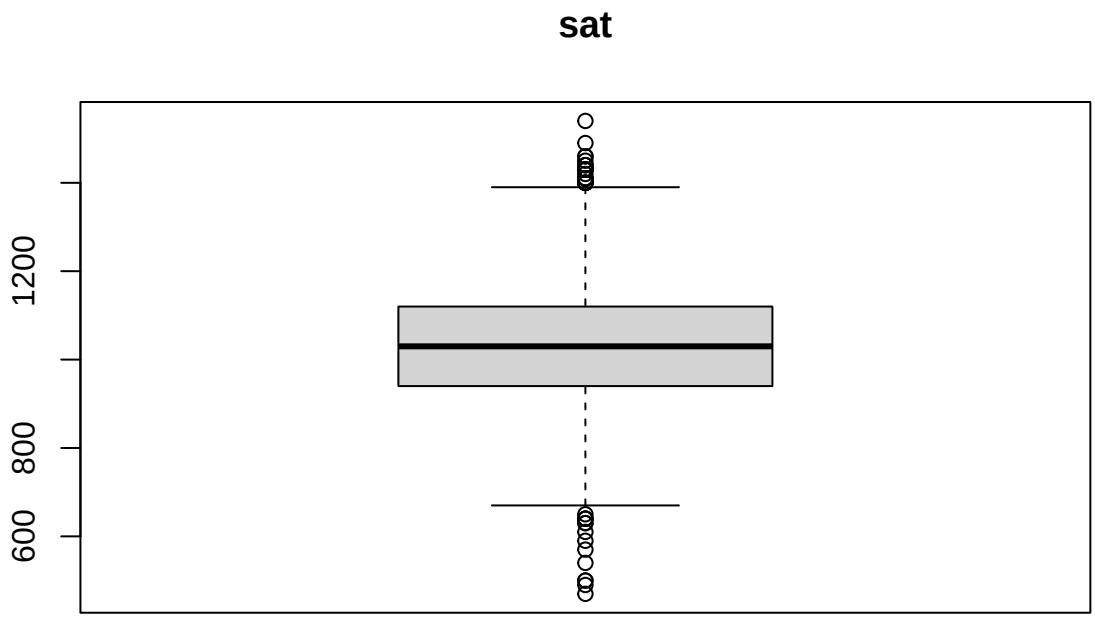
```
## integer(0)
```

```
# idx is null
```

4 Valores atípicos

Revisad si hay valores atípicos en las variables `sat` y `hsize`. Si se trata de un valor anómalo, es decir anormalmente alto o bajo, substituir su valor por NA y posteriormente, se imputará.

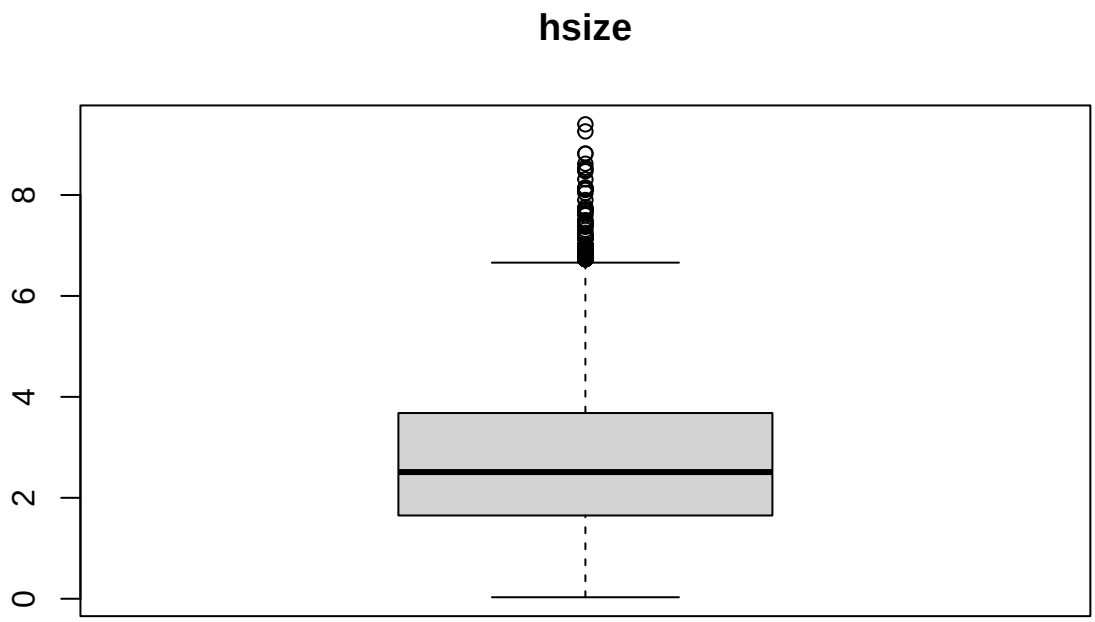
```
  #sat
  boxplot(ds$sat, main="sat")
```




```
  x<-boxplot.stats(ds$sat)$out
  idx <- which( ds$sat %in% x)
  sort(ds$sat[idx])
```

```
##  [1]  470  490  500  500  540  570  590  610  630  630  640  640  640  650 1400
## [16] 1400 1400 1400 1400 1410 1410 1410 1410 1420 1430 1430 1430 1430 1430
## [31] 1430 1440 1440 1440 1450 1460 1460 1490 1540
```

```
  #Los valores son correctos. No se  modifican.



  #hsize
  boxplot(ds$hsize, main="hsize")
```




```
  x<-boxplot.stats(ds$hsize)$out
  idx <- which( ds$hsize %in% x)
  sort(ds$hsize[idx])
```

```
##   [1] 6.73 6.73 6.73 6.73 6.73 6.73 6.73 6.73 6.73 6.75 6.75 6.75 6.75 6.77 6.80
##  [16] 6.82 6.82 6.82 6.82 6.82 6.82 6.86 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87
##  [31] 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.87 6.90 6.90
##  [46] 6.93 6.93 6.95 6.95 6.97 6.97 6.97 6.97 6.97 6.98 6.98 6.98 6.98 6.98 6.98
##  [61] 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98
##  [76] 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 6.98 7.00 7.00 7.00 7.00 7.00
##  [91] 7.00 7.00 7.00 7.01 7.01 7.03 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15
## [106] 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15 7.15
## [121] 7.15 7.15 7.18 7.18 7.18 7.18 7.18 7.18 7.20 7.21 7.25 7.31 7.37 7.40 7.40
## [136] 7.42 7.45 7.45 7.45 7.46 7.49 7.50 7.60 7.64 7.68 7.71 7.71 7.71 7.71 7.76
## [151] 7.90 8.04 8.10 8.12 8.12 8.12 8.12 8.12 8.12 8.15 8.31 8.47 8.50 8.54 8.62
## [166] 8.82 8.82 9.26 9.40
```

```
  #Los valores son correctos. No se  modifican.
```

5 Imputación de valores

Busque si hay valores perdidos en las variables cuantitativas. En el caso de detectar algún valor perdido es necesario realizar una imputación de valores en estas variables. Aplique imputación por vecinos más cercanos, utilizando la distancia de Gower, considerando en el cómputo de los vecinos más cercanos el resto de variables cuantitativas. Además, considere que la imputación debe realizarse con registros del mismo género. Por ejemplo, si un registro a imputar es mujer, se debe realizar la imputación usando sólo las variables

cuantitativas de los registros de mujeres.

Para realizar esta imputación, podéis usar la función "kNN" de la librería VIM con un número de vecinos igual a 11.

Mostrad que la imputación se ha realizado correctamente, mostrando el resultado de los datos afectados por la imputación.

```
# total registros
nrow(ds)
```

```
## [1] 4137
```

```
# Número de valors NA a cada variable

rx <- colSums(is.na(ds))
rx
```

```
##     sat  tothrs   hsize  hsrank  hsperc  colgpa athlete  female   white   black
##       0       0       0       0       0      41       0       0       0       0
```

```
# Total sin valores NAs
idx <- complete.cases(ds)
# Registros no completos
which(!idx)
```

```
##  [1]   40  100  318  343  490  500  629  846 1053 1172 1226 1238 1319 1450 1605
## [16] 1866 1888 1937 1975 2035 2108 2184 2530 2536 2691 2721 2728 2879 3149 3196
## [31] 3495 3496 3523 3546 3651 3660 3758 3798 3943 3998 4015
```

```
#Report of changes
info <- report(info,
               row=paste(which(!idx),collapse=", "),
               paste(sum(!idx),
                     "registres amb NA a la variable",
                     paste(names(which(rx>0)),collapse=", ")))


table(idx)
```

```
## idx
## FALSE  TRUE
##    41  4096
```

```
#Identificamos por separado los NAs de género femenino y los de género masculino
fem.idx <- which( is.na(ds$colgpa) & (ds$female=="TRUE") ); fem.idx
```

```
##  [1]  100  318  629 1172 1238 1319 1605 1866 1937 1975 2108 2530 2536 2721 2728
## [16] 3651 4015
```

```
mas.idx <- which( is.na(ds$colgpa) & ds$female=="FALSE"); mas.idx
```

```
##  [1]   40  343  490  500  846 1053 1226 1450 1888 2035 2184 2691 2879 3149 3196
## [16] 3495 3496 3523 3546 3660 3758 3798 3943 3998
```

```
#Imputamos registros female=="TRUE"
new.ds.fem<- kNN( ds[ ds$female=="TRUE", var.num], variable="colgpa", k=11)

new.ds.fem[new.ds.fem$colgpa==TRUE,]
```

```
## [1] sat        tothrs     hsize      hsrank     hsperc     colgpa     colgpa_imp
```

Cuadro 1: imputación valores colgpa mujeres

	sat	tothrs	hsize	hsrank	hsperc	colgpa
100	1120	49	0.10	1	10.000000	3.31
318	1050	12	3.70	30	8.108109	2.60
629	860	80	6.55	100	15.267180	2.82
1172	990	82	3.62	20	5.524862	2.81
1238	1060	120	4.39	32	7.289294	3.15
1319	1100	82	5.11	61	11.937380	2.55
1605	1030	77	5.70	108	18.947371	2.59
1866	810	84	2.14	10	4.672897	2.78
1937	830	46	4.50	97	21.555559	2.37
1975	1000	72	0.81	34	41.975311	2.19
2108	970	47	2.68	37	13.805970	2.72
2530	970	78	3.38	7	2.071006	2.74
2536	940	15	3.10	19	6.129032	2.78
2721	920	101	1.20	12	10.000000	2.67
2728	490	127	1.44	55	38.194439	2.60
3651	910	68	0.53	42	79.245293	2.41
4015	890	80	3.77	194	51.458889	2.46

```
## <0 rows> (or 0-length row.names)
```

```
ds[fem.idx,]$colgpa <- new.ds.fem[new.ds.fem$colgpa_imp==TRUE,]$colgpa

kable(ds[fem.idx, var.num],
      caption="imputación valores colgpa mujeres")
```

```
#Imputamos registros female=="FALSE"
new.ds.mas <- kNN( ds[ ds$female=="FALSE", var.num], variable="colgpa", k=11)
ds[mas.idx,]$colgpa <- new.ds.mas[new.ds.mas$colgpa_imp==TRUE,]$colgpa

kable(ds[mas.idx, var.num],
      caption="imputación valores colgpa hombres")
```

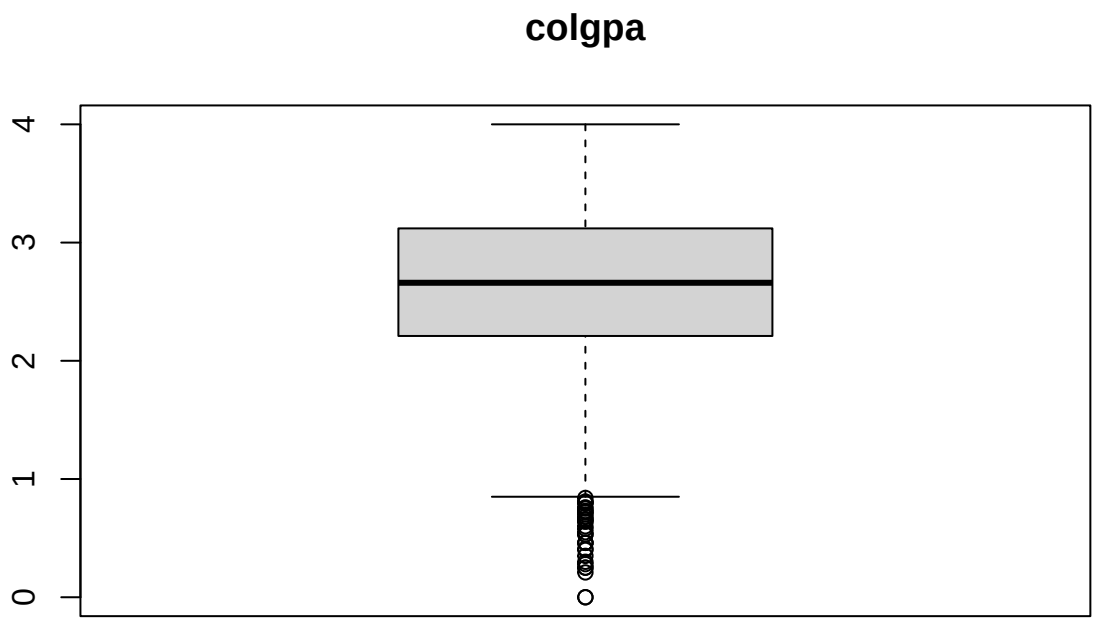
```
sum( complete.cases(ds$colgpa) )
```

```
## [1] 4137
```

```
boxplot( ds$colgpa, main="colgpa")
```

Cuadro 2: imputación valores colgpa hombres

	sat	tothrs	hsize	hsrank	hsperc	colgpa
40	940	19	1.85	41	22.162161	2.35
343	1100	43	7.45	218	29.261749	2.65
490	1090	95	4.72	67	14.194910	2.48
500	750	16	2.25	48	21.333330	2.18
846	970	39	2.21	77	34.841629	2.15
1053	900	17	1.54	44	28.571430	2.26
1226	1000	16	4.89	98	20.040899	2.72
1450	840	44	2.12	75	35.377361	2.26
1888	1040	16	0.83	23	27.710840	2.43
2035	990	78	0.72	3	4.166667	2.81
2184	1020	78	1.78	35	19.662920	2.60
2691	1260	50	7.00	47	6.714286	3.46
2879	1040	131	2.86	28	9.790210	2.69
3149	870	52	1.60	16	10.000000	2.50
3196	1070	14	2.34	36	15.384610	2.50
3495	910	13	4.89	145	29.652349	2.68
3496	910	40	4.77	125	26.205450	2.35
3523	1250	17	5.51	29	5.263158	2.76
3546	900	91	6.05	65	10.743800	2.42
3660	900	16	0.44	31	70.454536	2.26
3758	1160	120	1.72	11	6.395349	3.23
3798	930	44	2.92	6	2.054795	2.74
3943	1120	95	2.60	26	10.000000	3.41
3998	990	14	9.26	385	41.576679	1.68



6 Creación de una nueva variable

La variable `colgpa` contiene la nota numerica del alumnado. Crear una variable categorica denominada `gpaletter`, que indique la nota en letra de cada estudiante de la siguiente forma: A, de 3.50 a 4.00; B, de 2.50 a 3.49; C, de 1.50 a 2.49; D, de 0 a 1.49.

```
gpanum <- ds$colgpa
gpa_level<-c("D","C","B", "A")
classif <- ifelse( gpanum<=1.49, gpa_level[1],
                   ifelse(gpanum<=2.49, gpa_level[2],
                   ifelse(gpanum<=3.49, gpa_level[3],
                   gpa_level[4])))

ds$gpaletter <- factor( classif, order=TRUE, levels=gpa_level)

# checking
table(ds$gpaletter)
```

```
##
##    D    C    B    A
##  144 1536 1999  458
```

```
sum(table(ds$gpaletter))
```

```
## [1] 4137
```

```r
sum(table(ds$colgpa))
```

```
## [1] 4137
```

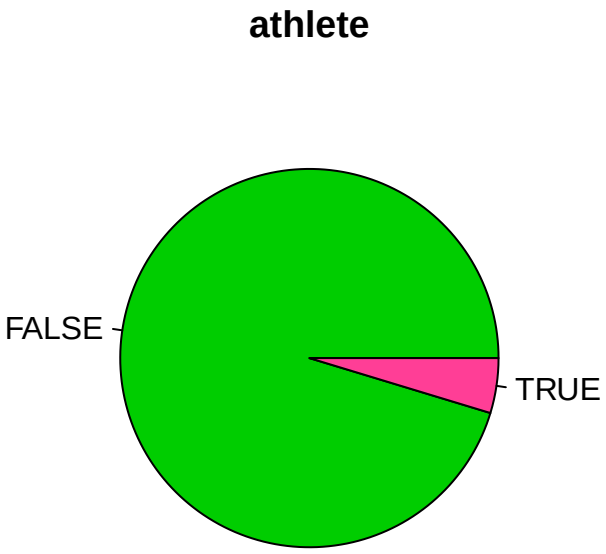
```r
#Report of changes

  info <- report(info, row="*", "gpaletter: Nueva variable que categoriza la nota numérica de colga en A
```

7 Estudio descriptivo

7.1 Estudio descriptivo de las variablas cualitativas

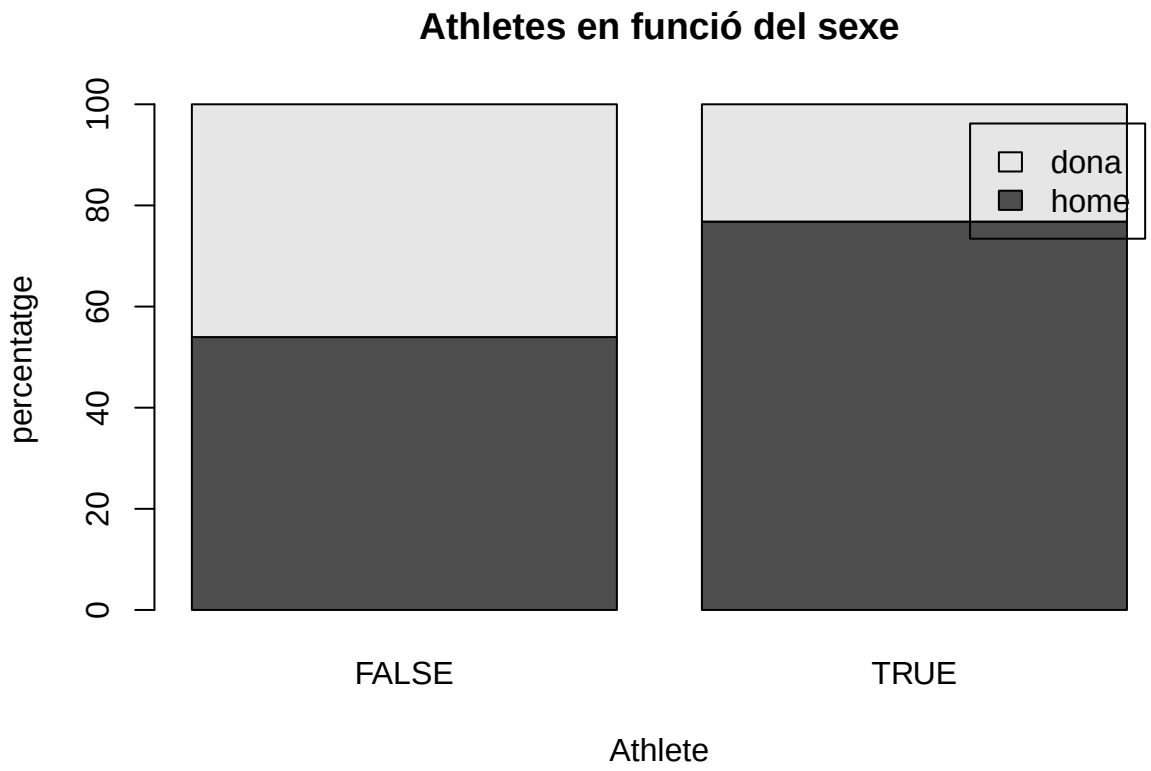
Represente en un primer gráfico, la variable `athlete` en porcentaje de atletas y un segundo gráfico, la variable `athlete` en función del sexo donde se muestre visualmente si el porcentaje de hombres y mujeres cambia al ser atleta o no.

```r
# pie plot
  pie(table(ds$athlete),
      main="athlete",
      col = c("green3", "violetred1")  )
```




```r
# bar plot a %
tb1 <- table(ds$female, ds$athlete)
tb2 <- prop.table(tb1,margin = 2)*100
barplot(tb2,
        xlab="Athlete",
        ylab= "percentatge",
```

```
        main = "Athletes en funció del sexe",
        legend = c("home", "dona")  )
```



7.2 Estudio descriptivo de las variables cuantitativas

Haga un estudio descriptivo de las variables cuantitativas "sat", "tothrs", "hsize", "hsrank".

Para ello, prepare una tabla con diversas medidas de tendencia central y dispersión, robustas y no robustas. Presente los gráficos donde se visualice la distribución de los valores de "sat" y "sat" en función del sexo.

```
idx.numeric <- which( colnames(ds) %in% c("sat", "tothrs", "hsize", "hsrank") )
mean.n   <- as.vector(sapply( ds[,idx.numeric ],mean,na.rm=TRUE ) )
std.n    <- as.vector(sapply(ds[,idx.numeric ],sd, na.rm=TRUE))
median.n <- as.vector(sapply(ds[,idx.numeric], median, na.rm=TRUE))
mean.trim.0.05 <- as.vector(sapply( ds[,idx.numeric],mean, na.rm=TRUE, trim=0.05))
mean.winsor.0.05 <- as.vector(sapply( ds[,idx.numeric], winsor.mean, na.rm=TRUE,trim=0.05))

IQR.n <- as.vector(sapply(ds[,idx.numeric],IQR, na.rm=TRUE))
mad.n <- as.vector(sapply(ds[,idx.numeric],mad, na.rm=TRUE))

kable(data.frame(variables= names(ds)[idx.numeric],
                 Media = mean.n,
                 Mediana = median.n,
                 Media.recort.0.05= mean.trim.0.05,
                 Media.winsor.0.05= mean.winsor.0.05
                 ),
      digits=2, caption="Estimaciones de Tendencia Central")
```

Cuadro 3: Estimaciones de Tendencia Central

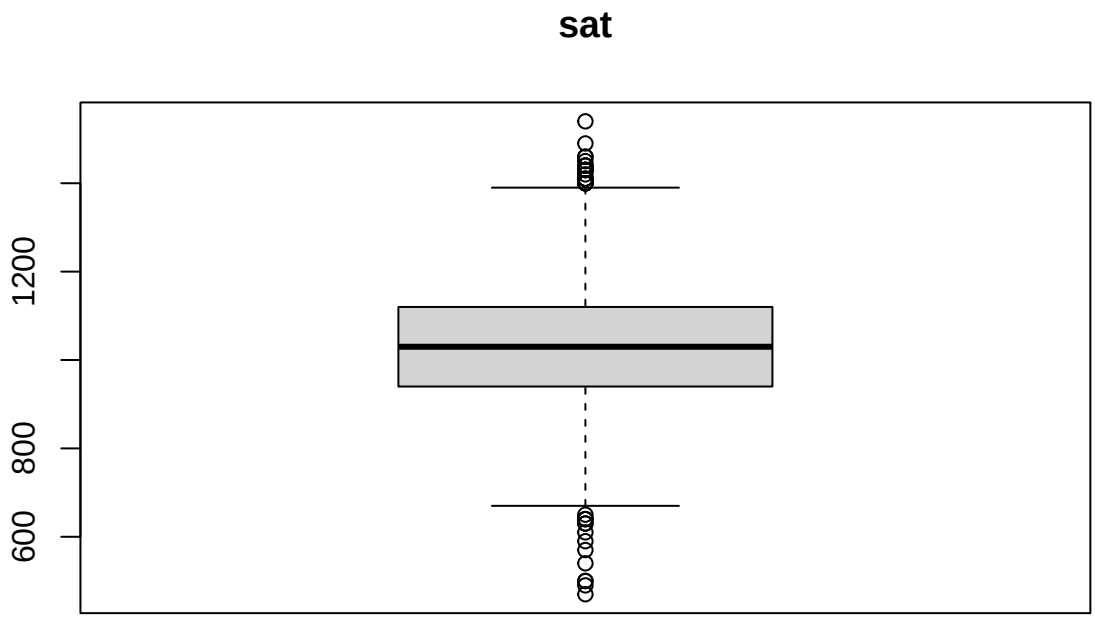
variables	Media	Mediana	Media.recort.0.05	Media.winsor.0.05
sat	1030.33	1030.00	1029.48	1030.53
tothrs	52.83	47.00	51.27	52.64
hsize	2.80	2.51	2.71	2.77
hsrank	52.83	30.00	43.99	48.78

Cuadro 4: Estimaciones de Dispersión

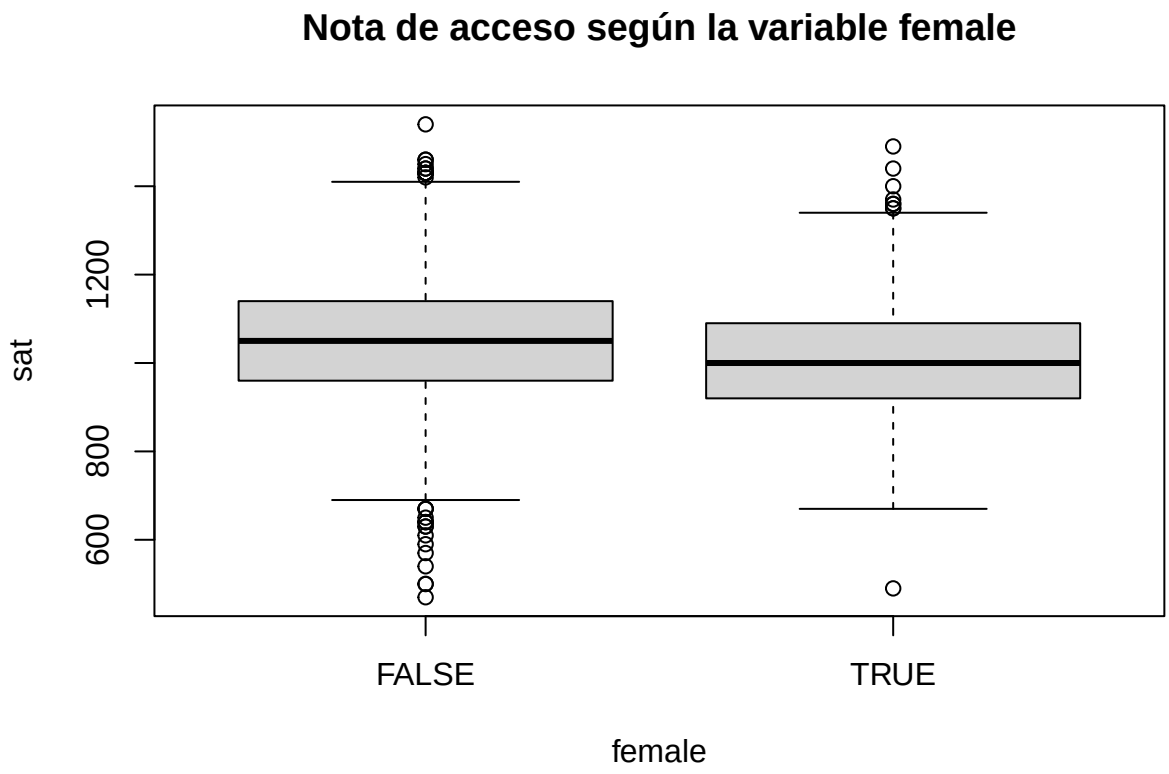
variables	Desv.Standard	IQR	MAD
sat	139.40	180.00	133.43
tothrs	35.33	63.00	45.96
hsize	1.74	2.03	1.42
hsrank	64.68	59.00	35.58

```
kable(data.frame(variables= names(ds)[idx.numeric],
                 Desv.Standard = std.n,
                 IQR = IQR.n,
                 MAD = mad.n
                 ),
      digits=2, caption="Estimaciones de Dispersión")
```

```
#Plots
boxplot(ds$sat, main="sat")
```




```
boxplot(sat ~ female, data= ds, main="Nota de acceso según la variable female")
```



8 Archivo final

Una vez realizado el preprocesamiento sobre el archivo, copiad el resultado de los datos en un archivo llamado **gpa_clean.csv**.

```
info <- report(info,row="",
                   message= paste0("n. row = ", nrow(ds),"; ",
                                   "n. col= ", ncol(ds), "; ",
                                   "n. var num. = ",  length(id.num),"; ",
                                   "n. var cualit. = ", length(id.factor)+1))


write.csv(ds, "gpa_clean.csv", row.names = FALSE)
```

9 Informe ejecutivo

Está formado por dos partes:

- Tabla resumen de los cambios realizados en el preprocesamiento.
- Breve explicación de las características estadísticas básicas de cada variable por separado.

9.1 Tabla resumen del preprocesamiento

Documentar de forma resumida, en forma de tabla, los cambios introducidos en el archivo original durante su preprocesamiento. Hay que explicar el detalle del preproceso aplicado. Por ejemplo, no basta con decir "se ha normalizado la variable hsize". En todo caso, debería indicarse si se ha reemplazado la coma por el punto decimal, o si se han redondeado decimales, etcétera y a que observaciones. Debe ser específicos, ya que el informe a de ser útil como documentación de los cambios realizados.

La primera y última fila de la tabla debe indicar el número de observaciones, el número de variables cuantitativas, el número de variables cualitativas y el total de variables al inicio del preprocesamiento y al final, respectivamente.

```
info <- info[1:nrow(info),]
info %>%
  kable( caption="Resumen del preproceso", row.names = FALSE)  %>%
  column_spec(2:3, width = "20em") %>%
  kable_styling( latex_options=c("striped", "repeat_"))
```

Cuadro 5: Resumen del preproceso

id	row	message
1		n. row = 4137; n. col= 10; n. var num. = 6; n. var cualit. = 4
1	65, 385, 720, 939, 1293, 1312, 2330, 2400, 2933, 3193, 3543	athlete: false -> FALSE
2	876	athlete: true -> TRUE
3	307, 754, 1230, 1858, 2213, 2374, 2376, 3042, 3173	black: Eliminado espacios en blanco
4	1255, 1785, 2227, 2424, 2450, 2913, 3076, 3102, 3803, 3868	black: false -> FALSE
5	457, 595, 956, 2100, 3787, 3854	white: Eliminar espacios en blanco
6	1929, 2922, 3536	white: false -> FALSE
7	461, 922, 1007, 1947, 2810, 2969, 3129, 4029, 4030	white: true -> TRUE
8	*	tothrs: Eliminado el texto h y seha convertido a variable numérica
9	53, 67, 155, 214, 371, 557, 565, 784, 842, 911, 948, 1399, 1566, 1723, 1956, 2024, 2293, 2304, 2361, 2382, 2603, 2689, 3832, 4003	corregimos la coma por el punto decimal
10	188, 201, 313, 657, 876, 2489, 3438, 3441, 3445, 3537, 3753, 4091	hsperc: recalcular los valores de hsrank/hsize
11	40, 100, 318, 343, 490, 500, 629, 846, 1053, 1172, 1226, 1238, 1319, 1450, 1605, 1866, 1888, 1937, 1975, 2035, 2108, 2184, 2530, 2536, 2691, 2721, 2728, 2879, 3149, 3196, 3495, 3496, 3523, 3546, 3651, 3660, 3758, 3798, 3943, 3998, 4015	41 registres amb NA a la variable colgpa
12	*	gpaletter: Nueva variable que categoriza la nota numérica de colga en A, de 3.50 a 4.00; B, de 2.50 a 3.49; C, de 1.50 a 2.49; D, de 0 a 1.49
13		n. row = 4137; n. col= 11; n. var num. = 6; n. var cualit. = 5

9.2 Resumen estadístico

A partir de la información obtenida en los apartados anteriores haga un breve comentario de cada variable (2 o tres lineas) destacando el más relevante y característico. El resumen no debe ocupar más de una página.

- **sat**: Variable numérica. Representa la nota de acceso a la Universidad. Los valores de media, mediana y medianas trimmed y winsor son muy similares, alrededor de 1030. Esto indica que la distribución de los datos es prácticamente simétrica. Respecto a las estimaciones de dispersión, tienen valores similares desv. estándar y MAD, el IQR tiene un valor mayor de 180.
- **tothrs**: Variable numérica. Representa el total de horas cursadas en el semestre. Los valores de media están en torno a 52 horas, en cambio la media es algo menor, 47 horas. Esto significa que hay estudiantes con valores muy altos que aumentan el valor de la media respecto a la mediana. Los valores de dispersión varían. Así se tiene que la desv. estándar tiene el valor de 35.33 hasta el IQR que vale 63.
- **hsize**: Variable numérica. Representa el número total de estudiantes en la cohorte de graduados del bachillerato (en cientos). Los valores de las medias son bastante similares entre 2.71 a 2.80. Por el contrario, el valor de mediana baja algo más a 2.51. Es el efecto de algunos valores extremos. Respecto a la dispersión, se mueve muy poco entre un valor de MAD de 1.42 hasta el IQR de 2.03.
- **hsrank**: Variable numérica. Ranking del estudiante dado por la nota media del bachillerato de la cohorte de graduados del bachillerato. Esta variable es la que presenta mayor diferencia entre los valores de media y mediana en comparación con otras variables, de unos 23 puntos. Esto indica que la variable tiene algunos valores extremos. La mediana es de 30 puntos, mientras que la media tiene un valor de 52.83. En cualquier caso, como esta variable va asociada a `hsize` es normal que si un instituto tiene muchos alumnos pueda tener valores muy altos de `hsrank`. En cambio, en institutos pequeños la mayoría de los valores de `hsize` quedarán en posiciones bajas. Esta variabilidad también queda reflejada con las estimaciones de dispersión que van de 35.58 para MAD hasta 64.68 por la desv. estándar.
- **hsperc**: Variable numérica. Ranking relativo del estudiante. Como esta variable normaliza `hsrank` en función del tamaño del instituto `hsize` no tiene esta diferencia tan extrema entre los valores de media y mediana y, los valores de estimaciones de dispersión como `hsrank`. Aunque existen algunos valores extremos.
- **colgpa**: Variable numérica. Representa la nota media al final del primer semestre. Variable que tiene una media y mediana muy similares. Viendo el boxplot se observa que la distribución de valores es asimétrica con una cola a la izquierda desde cero hasta alrededor de 2.4 que después baja hasta 4.
- **athlete**: Variable categórica binaria. Distribución muy desigual, la mayoría de los estudiantes (95%) no practica ningún deporte en la universidad.
- **female**: Variable categórica binaria. Distribución muy similar con un porcentaje ligeramente superior de estudiantes varones (55,04%) respecto a mujeres (44,96%).
- **white**: Variable categórica binaria. Distribución muy desigual. La mayoría son white (92,55%)
- **black**: Variable categórica binaria. Distribución muy desigual. Sólo un 5,55% de los alumnos son black. Si consideramos la información de las variables `white` y `black` conjuntamente podemos observar que existen unos 79 alumnos que no son ni black, ni white. Una proporción muy pequeña (0,02%).
- **gpaletter**: Variable categórica con cuatro categorías. La distribución de valores es D (3.48%), C(37.13%), B(48.32%) y A (11.07%). Así que la mayoría de los alumnos aprueban y una parte importante de éstos con unas notas altas.

10 Evaluación de la actividad

- Secciones 1, 2 (10%)
- Secciones 3, 4 (20%)

- Sección 5 (10%)
- Sección 6 (10%)
- Secciones 7, 8 (20%)
- Sección 9 (20%)
- Calidad del informe dinámico (calidad del código, formato y estructura del documento, concisión y precisión en las respuestas) (10%)

11 Referencias

Quick-R

Cookbook for R

LaTex tables

Data Visualization with R