PRA2 - Tipología y ciclo de vida de los datos

Lingfeng Zheng

Contents

1. Descripción del dataset

Se ha elegido el siguiente conjunto de datos: gpa_row.csv. Por se un conjunto de datos que contiene una amplia variedad de datos numéricos y categóricos para poder realizar un análisis rico y sacar conclusiones a una serie de preguntas.

Este es un dataset que 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.

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 conjunto de datos pasando por las diferentes etapas de tratamiento de datos, para dejar un conjunto de datos listo 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.

Con el análisis se pretenderá responder las siguientes preguntas:

- ¿Qué variables cuantitativas influyen más en la nota?
- ¿Ser atleta influye en la nota?
- ¿Las mujeres tienen mejor nota que los hombres?

---

2. Integración y selección

De las variables presentadas anteriormente se procede a eliminar la variable hsperc, al ser esta una variable derivada que se calcula a partir de las otras dos presentes en el mismo dataset.

Como en el desarrollo de esta practica hay una parte que trata sobre la creación de un modelo para el análisis de regresión, tener más variables a la hora de crear un modelo no siempre conlleva a tener un modelo mejor, sino que puede generar ruidos y empeorar el modelo. Especialmente en este caso, que se trata de una variable derivada de otras dos presentes en el conjunto de datos.

Por lo que se procede a eliminar este variable.

Primero se procede a la lectura del conjunto de datos:

```
gpa_raw <-read.csv("gpa_row.csv" ,sep=",", stringsAsFactors=TRUE)
gpa <- gpa_raw[,-5]
```

---

3. Limpieza de los datos

```
# Isnpeccionamos la dimensión del dataset
dim(gpa)
```

```
## [1] 4137    9
```

```
# Vemos cómo ha interpretado cada columna al cargar el csv
str(gpa)
```

```
## 'data.frame':    4137 obs. of  9 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 ...
##  $ 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 ...
```

Vamos a aplicar primero normalización sobre las variables, atendiendo a estos criterios:

- 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.
- 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.

```
#aplicamos toupper para trasnformarlos todos a mayúscula
gpa$athlete <- toupper(gpa$athlete)
#lo transformamos de nuevo al tipo factor
gpa$athlete <- as.factor(gpa$athlete)
```

```
#lo transformamos a tipo factor
gpa$female <- as.factor(gpa$female)
```

```
#aplicamos toupper y trimws, y lo transformamos a factor de nuevo
gpa$black <- trimws(toupper(gpa$black))
gpa$black <- as.factor(gpa$black)
```

```
gpa$white <- trimws(toupper(gpa$white))
gpa$white <- as.factor(gpa$white)
```

```
gpa$sat <- as.numeric(gpa$sat)
```

```
#convertir tipo facotr a caracter
hours <- as.character(gpa$tothrs)
#aplicar word que devuelve el primer caracter y con h como separador, para sacar la parte numerica
library(stringr)
val <- word(hours,sep=fixed("h"))
# convertir en integer
gpa$tothrs <- as.numeric(gpa$tothrs)
```

```
#aplicar cambio de coma por punto y comprobar que no hay más valores coma decimal
gpa$hsize <- gsub("\\,","\\.",gpa$hsize)
coma_sep <- grep("\\,", gpa$hsize)
coma_sep
```

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

```
#transformar a tipo numeric y redondear a 2 unidad decimal como pide el enunciado
gpa$hsize <- as.numeric(gpa$hsize)
gpa$hsize <- round(gpa$hsize, digits = 2)
```

```
# Comprobamos el resultado tras aplicar la normalización
str(gpa)
```

```
## 'data.frame':    4137 obs. of  9 variables:
##  $ sat    : num  920 1170 810 940 1180 980 880 980 1240 1230 ...
##  $ tothrs : num  67 46 42 64 46 16 103 79 46 45 ...
##  $ hsize  : num  0.1 9.4 1.19 5.71 2.14 2.68 3.11 2.68 3.67 0.1 ...
##  $ hsrank : int  4 191 42 252 86 41 161 101 161 3 ...
##  $ colgpa : num  2.04 4 1.78 2.42 2.61 ...
##  $ athlete: Factor w/ 2 levels "FALSE","TRUE": 2 1 2 1 1 1 1 1 1 1 ...
##  $ female : Factor w/ 2 levels "FALSE","TRUE": 2 1 1 1 1 2 1 1 1 1 ...
##  $ white  : Factor w/ 2 levels "FALSE","TRUE": 1 2 2 2 2 2 1 2 2 2 ...
##  $ black  : Factor w/ 2 levels "FALSE","TRUE": 1 1 1 1 1 1 1 1 1 1 ...
```

3.1 ¿Los datos contienen ceros o elementos vacíos? Gestiona cada uno de estos casos.

Primero comprobamos que columnas contiene valores perdidos:

```
names(which(colSums(is.na(gpa))>0))
```

```
## [1] "colgpa"
```

Vemos que hay valores perdidos para la variable colgpa, vamos a proceder a imputar estos valores perdidos aplicando la tecnica de 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, para tener mayor calidad a la hora de aplicar la imputación, se realizará de forma que la imputación se realizará con registros del mismo género. Es decir, si un registro a imputar es mujer, se debe realizar la imputación usando sólo las variables cuantitativas de los registros de mujeres.

Para ello, usaremos la función KNN:

```
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("VIM")
```

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

```
## VIM is ready to use.
```

```
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
```

```
##
## Attaching package: 'VIM'
```

```
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
gpa_fem_imput <- kNN( filter(gpa,female=="TRUE"), variable= "colgpa", k= 11)
gpa_male_imput <- kNN( filter(gpa,female=="FALSE"), variable= "colgpa", k= 11)
fem_na_subset <- which(gpa$female=="TRUE" & is.na(gpa$colgpa))
male_na_subset <- which(gpa$female=="FALSE" & is.na(gpa$colgpa))
```

```
head(gpa[fem_na_subset,])
```

```
##       sat tothrs hsize hsrank colgpa athlete female white black
## 100  1120     73  0.10      1     NA   FALSE   TRUE  TRUE FALSE
## 318  1050     33  3.70     30     NA   FALSE   TRUE  TRUE FALSE
## 629   860    105  6.55    100     NA   FALSE   TRUE  TRUE FALSE
## 1172  990    107  3.62     20     NA   FALSE   TRUE  TRUE FALSE
## 1238 1060     23  4.39     32     NA   FALSE   TRUE  TRUE FALSE
```

```
## 1319 1100    107  5.11     61     NA   FALSE   TRUE  TRUE FALSE
```

```
head(gpa[male_na_subset,])
```

```
##       sat tothrs hsize hsrank colgpa athlete female white black
## 40    940     47  1.85     41     NA   FALSE  FALSE  TRUE FALSE
## 343  1100     67  7.45    218     NA   FALSE  FALSE  TRUE FALSE
## 490  1090    120  4.72     67     NA    TRUE  FALSE  TRUE FALSE
## 500   750     44  2.25     48     NA   FALSE  FALSE  TRUE FALSE
## 846   970     63  2.21     77     NA   FALSE  FALSE  TRUE FALSE
## 1053  900     45  1.54     44     NA   FALSE  FALSE  TRUE FALSE
```

```
# Imputar por los valores salida de KNN
gpa[fem_na_subset,]$colgpa <- gpa_fem_imput[gpa_fem_imput$colgpa_imp==TRUE,]$colgpa
gpa[male_na_subset,]$colgpa <- gpa_male_imput[gpa_male_imput$colgpa_imp==TRUE,]$colgpa
```

```
# Comprobamos que se han imputado correctamente los valores perdidos
head(gpa[fem_na_subset,])
```

```
##       sat tothrs hsize hsrank colgpa athlete female white black
## 100  1120     73  0.10      1   2.98   FALSE   TRUE  TRUE FALSE
## 318  1050     33  3.70     30   2.58   FALSE   TRUE  TRUE FALSE
## 629   860    105  6.55    100   2.87   FALSE   TRUE  TRUE FALSE
## 1172  990    107  3.62     20   2.93   FALSE   TRUE  TRUE FALSE
## 1238 1060     23  4.39     32   2.91   FALSE   TRUE  TRUE FALSE
## 1319 1100    107  5.11     61   2.94   FALSE   TRUE  TRUE FALSE
```

```
# Comprobamos que se han imputado correctamente los valores perdidos
head(gpa[male_na_subset,])
```

```
##       sat tothrs hsize hsrank colgpa athlete female white black
## 40    940     47  1.85     41   2.43   FALSE  FALSE  TRUE FALSE
## 343  1100     67  7.45    218   2.65   FALSE  FALSE  TRUE FALSE
## 490  1090    120  4.72     67   2.50    TRUE  FALSE  TRUE FALSE
## 500   750     44  2.25     48   2.62   FALSE  FALSE  TRUE FALSE
## 846   970     63  2.21     77   2.21   FALSE  FALSE  TRUE FALSE
## 1053  900     45  1.54     44   2.42   FALSE  FALSE  TRUE FALSE
```

3.2 Identifica y gestiona los valores extremos.

En este paso vamos a analizar los valores extremos de las variables numéricas y ver si hay valores atípicos . En caso afirmativo, analizar si se trata de valores anómalos.

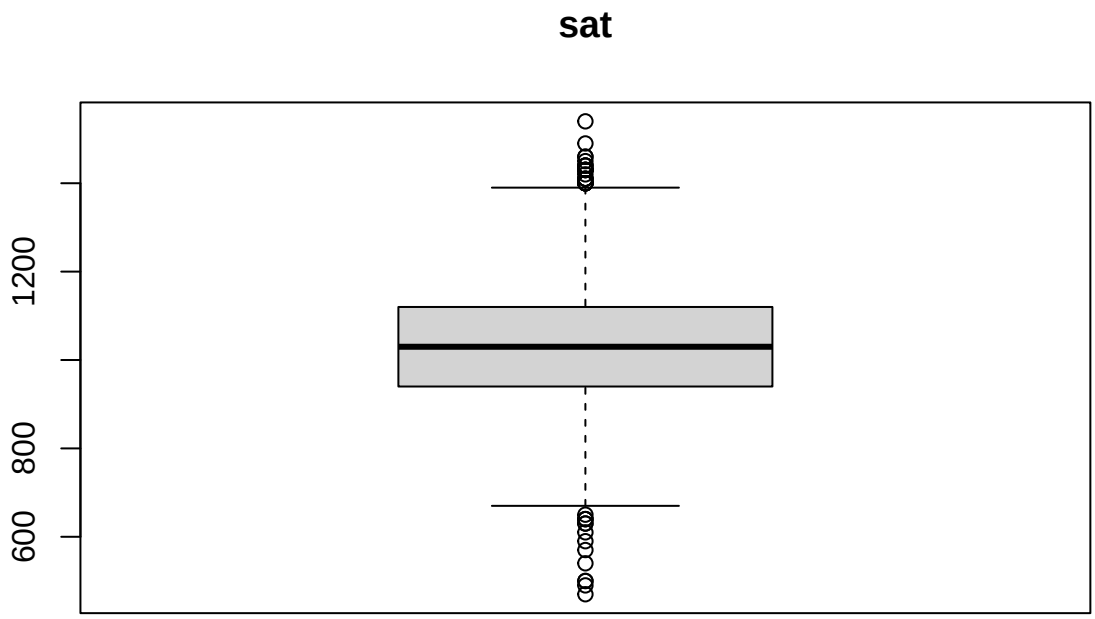
En caso de que hay presente valores anómalos, el procedimiento sería de sustituir estos por NA y aplicar un metodo de imputación para esos valores.

Los valores extremos son aquellos que presentan valores no congruentes comparados con el resto de los datos, por ejemplo los valores que aparecen en los extremos del rango intercuartílico.

Para ello, vamos a usar las gráficas de caja para analizar los valores atípicos.

Los valores atípicos son los que están en los estremos, fuera del límite de 25 percentil o 75 percentil. Representados como circulos blancos en las gráficas de cajas:

```
boxplot(gpa$sat, main="sat")
```

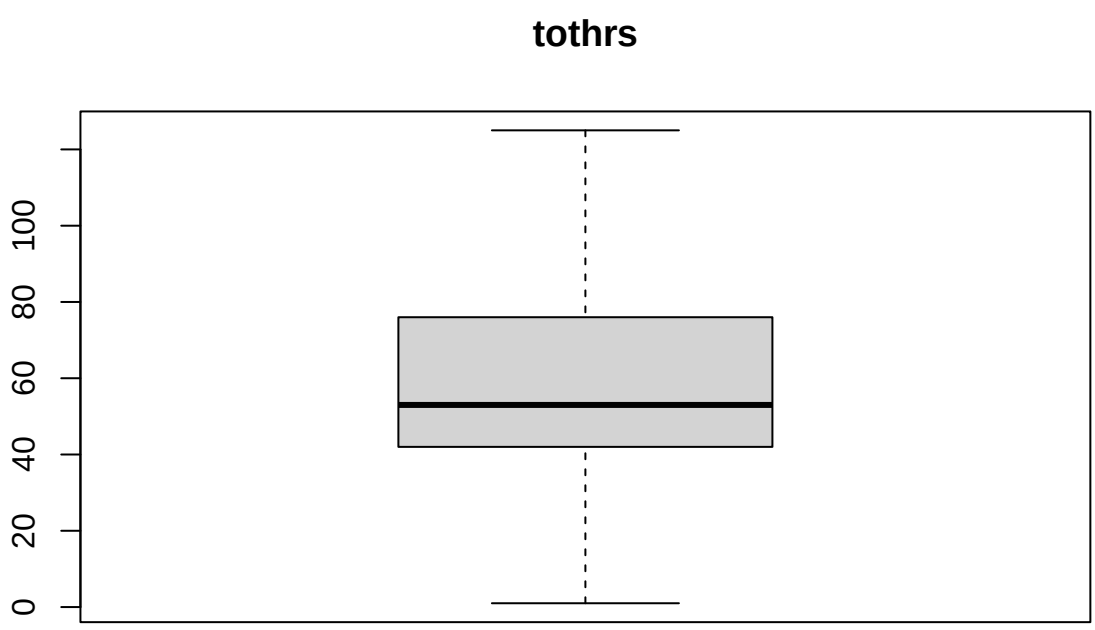



```
summary(gpa$sat)
```

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

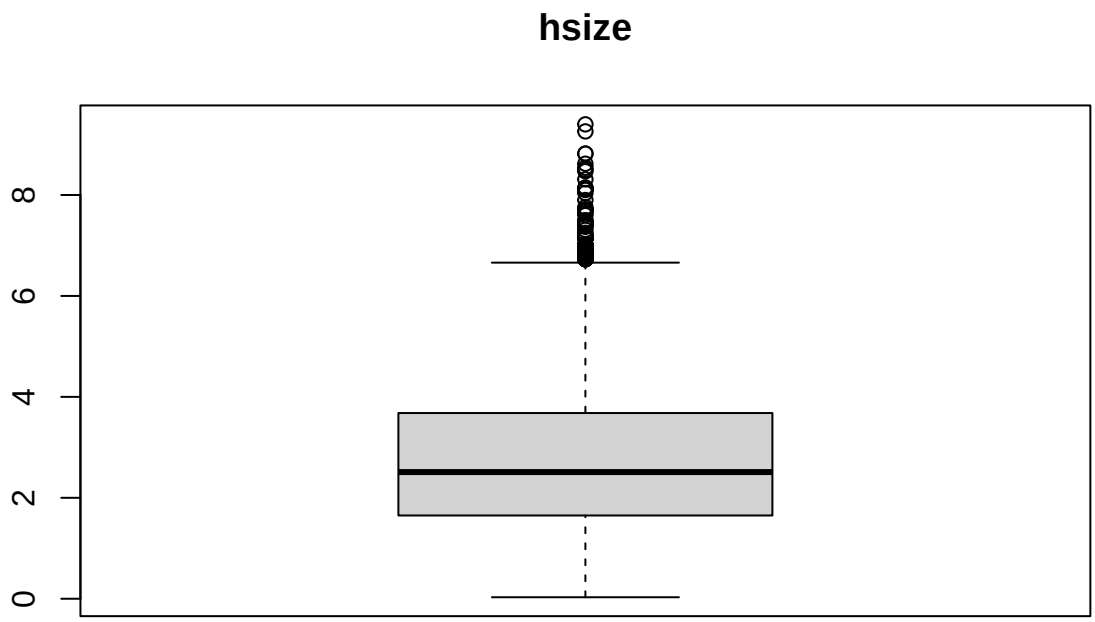
La variable sat tiene valores atípicos según la gráfica. Pero no valores anómalos, al todos los valores dentro del rango es de 400 a 1600.

```
boxplot(gpa$tothrs,main="tothrs")
```



La variable tothr no tiene valores atípicos.

```
boxplot(gpa$hsize,main="hsize")
```

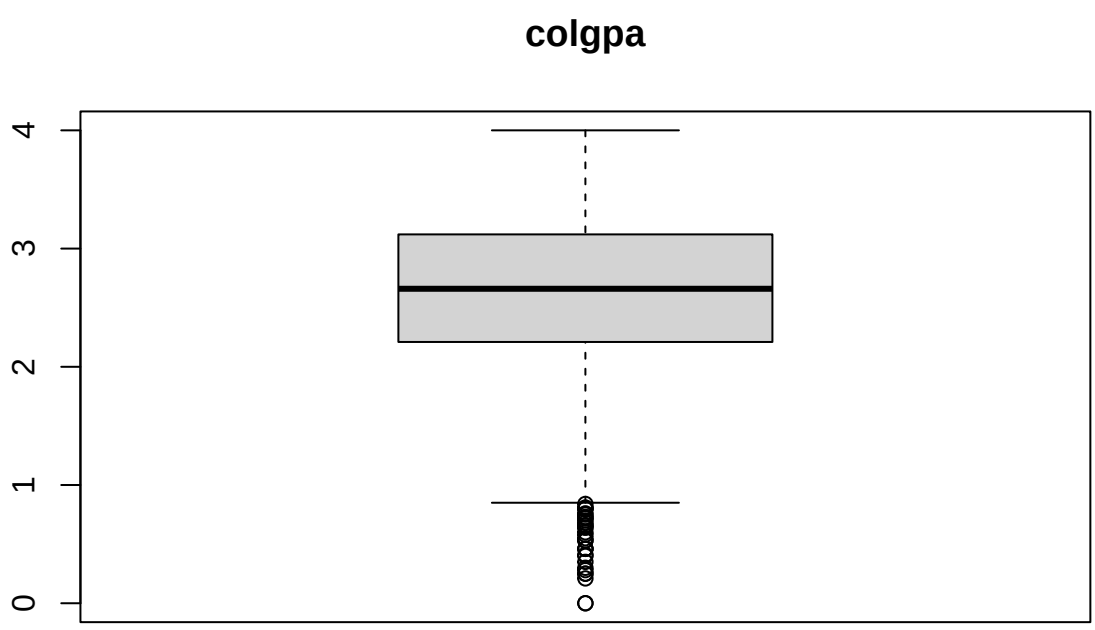



```
summary(gpa$hsize)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.03    1.65    2.51    2.80    3.68    9.40
```

La variable hsize tiene valores atípicos, pero tampoco son valores anómalos. Ya que la variable hsize representa el número total de estudiantes en la cohorte (en cientos), de manera que significa que tiene un número de alumnos que va de 3 a 940.

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




```
summary(gpa$colgpa)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.210   2.660   2.654   3.120   4.000
```

La variable colgpa tiene valores atípicos según la gráfica. Pero no son valores anómalos, ya que todos los valores están dentro del rango permitido de 0.0 a 4.0.

---

4 Análisis de los datos

4.1 Selección de los grupos de datos que se quieren analizar/comparar (p. ej., si se van a comparar grupos de datos, ¿cuáles son estos grupos y qué tipo de análisis se van a aplicar?)

De acuerdo con las preguntas planteadas al inicio.

Se va a separar en grupos según si es hombre o mujer:

```
gpa_female<-gpa$colgpa[gpa$female=="TRUE"]
gpa_male<-gpa$colgpa[gpa$female=="FALSE"]
```

Según si es atleta o no:

```
gpa_athlete<-gpa$colgpa[gpa$white=="TRUE"]
gpa_no_athlete<-gpa$colgpa[gpa$black=="FALSE"]
```

4.2 Comprobación de la normalidad y homogeneidad de la varianza.

Vamos a utilizar el test de normalidad de Shapiro-Wilk para comprobar la normalidad y homogeidad de la varianza.

El test de normalidad de Shapiro-Wilk trabaja con la hipótesis nula de normalidad de los datos. Para los valores de p del test inferiores al nivel de significancia permiten rechazar la hipótesis nula y, por lo tanto, llevarían a descartar la normalidad de los datos.

```
alpha = 0.05
col_names = colnames(gpa)
for (i in 1:ncol(gpa)) {
  if (i == 1) cat("Las variables que no siguen una distribución normal son:\n")
  if (is.integer(gpa[,i]) | is.numeric(gpa[,i])) {
    p_val = shapiro.test(gpa[,i])$p.value
    if (p_val < alpha) {
      cat(col_names[i])
      # Format output
      if (i < ncol(gpa) - 1) cat(", ")
      if (i %% 3 == 0) cat("\n")
    }
  }
}
```

```
## Las variables que no siguen una distribución normal son:
## sat, tothrs, hsize,
## hsrank, colgpa,
```

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.

4.3.1 ¿Ser atleta influye en la nota?

En este apartado queremos analizar si ser atleta influye en la nota colgpa. Es decir, si hay diferencias significativas entre atletas y no atletas en esta nota, con un nivel de confianza del 95%.

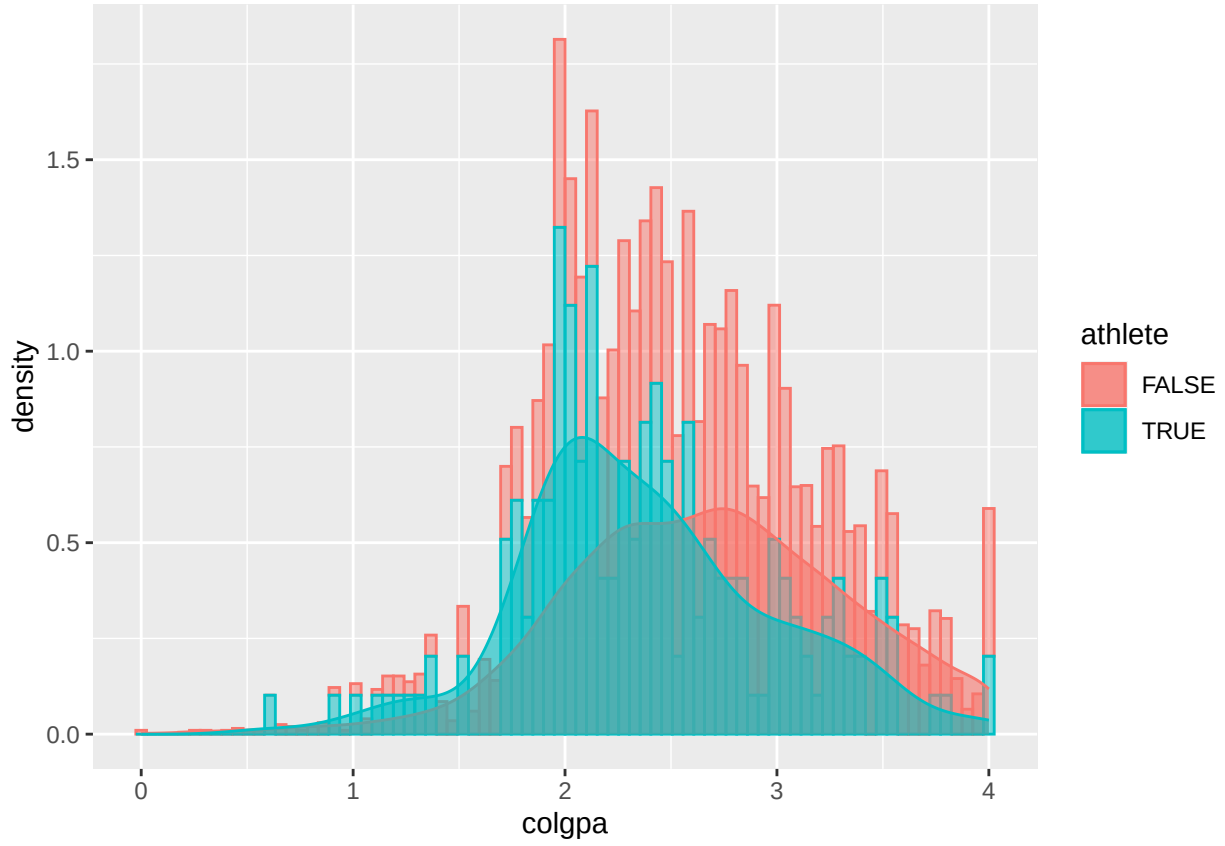
```

library(ggplot2)
```

4.3.1.1 Análisis visual:

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```
ggplot(gpa, aes(x=colgpa, color=athlete, fill=athlete)) +
  geom_histogram( aes(y=..density..), bins = 80,  alpha=0.5)+
  geom_density( alpha=0.6)
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
```



4.3.1.2 Hipótesis nula y la alternativa En este caso se trata de una comparación de medias en poblaciones independientes:

H0 : $\mu_{\text{athlete}} = \mu_{\text{no\_athlete}}$

H1 : $\mu_{\text{athlete}} \neq \mu_{\text{no\_athete}}$

4.3.1.3 Justificación del test a aplicar Necesitamos aplicar el test de dos muestras sobre la media con varianzas desconocidas.

Como en este caso, el número de muestras es mayor que 30, podemos aplicar el teorema del límite central y considerar la aproximación a la distribución normal.

Lo siguiente es comprobar la igualdad o no de varianzas:

```
ath_true<-gpa$colgpa[gpa$athlete=="TRUE"]
ath_false<-gpa$colgpa[gpa$athlete=="FALSE"]
var.test(ath_true,ath_false)
```

```
##
##  F test to compare two variances
##
## data:  ath_true and ath_false
## F = 0.82169, num df = 193, denom df = 3942, p-value = 0.07235
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.6759629 1.0182486
## sample estimates:
```

```
## ratio of variances
##          0.8216948
```

El resultado del var.test anterior nos devuelve un valor de 0.07235 para p, que es mayor que 0.05. Por lo que no podemos rechazar la hipóteis nula de igualdad de varianzas, es decir, tienen misma varianza.

```
t.test(ath_true,ath_false,var.equal=TRUE)
```

4.3.1.4 Interpretación del test

```
##
##  Two Sample t-test
##
## data:  ath_true and ath_false
## t = -5.9506, df = 4135, p-value = 2.893e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3812574 -0.1922910
## sample estimates:
## mean of x mean of y
##  2.380464  2.667238
```

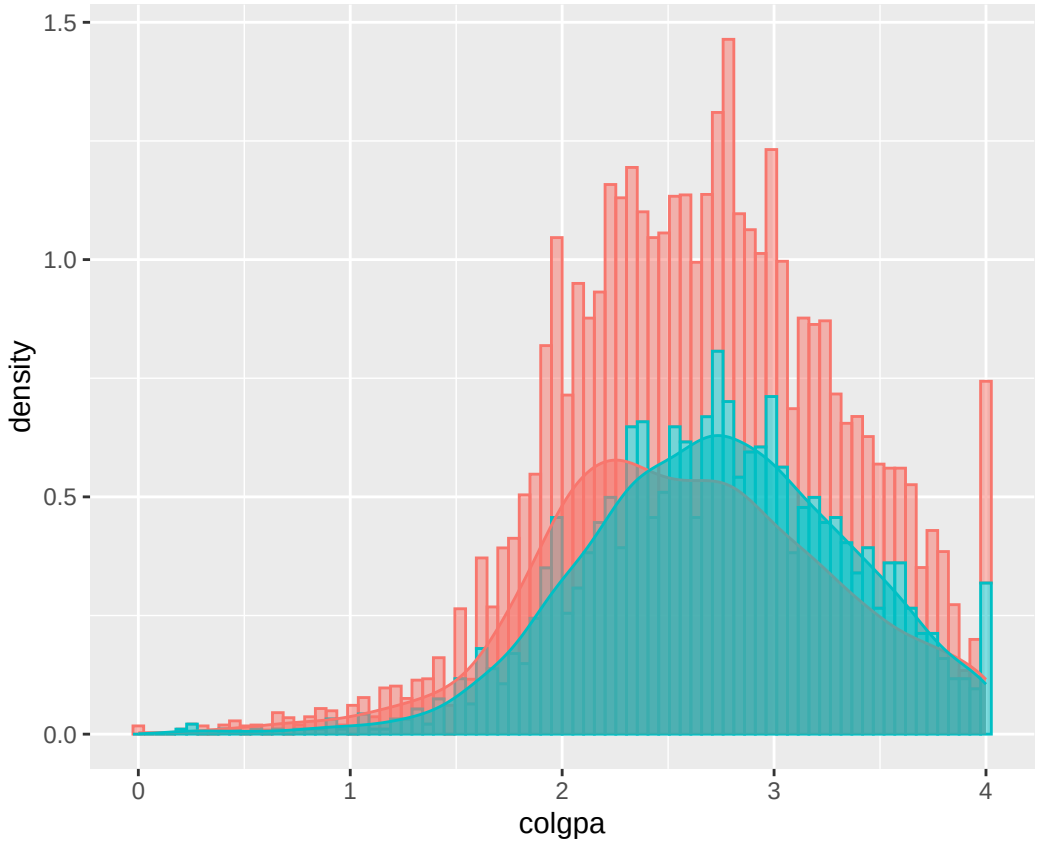
El pvalor del test (3.69 $\times$ 10-9) es inferior al nivel de signifificación (0.05). Además el valor observado -5.91 no se encuentra dentro de la zona de aceptación.

Por tanto, podemos rechazar la hipótesis nula a favor de la alternativa y podemos concluir que en promedio la nota media de los alumnos que practican el deporte es diferente de la de los alumos que no practican el deporte.

4.3.2 ¿Las mujeres tienen mejor nota que los hombres?

En este apartado queremos analizar si el género influye en la nota colgpa. Específicamente, si las mujeres tienen mejor nota que los hombres, con un nivel de confianza del 95%.

```
library(ggplot2)
ggplot(gpa, aes(x=colgpa, color=female, fill=female)) +
  geom_histogram( aes(y=..density..), bins = 80,  alpha=0.5)+
  geom_density( alpha=0.6)
```



4.3.2.1 Análisis visual:

4.3.2.2 Hipótesis nula y la alternativa En este caso se trata de una comparación de medias en poblaciones independientes:

H0 : $\mu_{\text{female}} = \mu_{\text{male}}$

H1 : $\mu_{\text{female}} > \mu_{\text{male}}$

4.3.2.3 Justificación del test a aplicar Necesitamos aplicar el test de dos muestras sobre la media con varianzas desconocidas.

Como en este caso, el número de muestras es mayor que 30, podemos aplicar el teorema del límite central y considerar la aproximación a la distribución normal.

Lo siguiente es comprobar la igualdad o no de varianzas:

```
var.test(gpa_female,gpa_male)
```

```
##
##  F test to compare two variances
##
## data:  gpa_female and gpa_male
## F = 0.82663, num df = 1859, denom df = 2276, p-value = 1.804e-05
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.7581404 0.9016441
## sample estimates:
## ratio of variances
##          0.8266259
```

El resultado del var.test anterior nos devuelve un valor de 1.804 × 10-5 para p, que es menor que 0.05. Por lo que podemos rechazar la hipóteis nula de igualdad de varianzas, es decir, tienen distinta varianza.

Por último, concluimos que el tipo de test sería un test de dos muestras sobre la media con varianza desconocida y distinta. Y sería un test unilateral por la derecha.

```
t.test(gpa_female,gpa_male,var.equal=FALSE, alternative = "greater")
```

4.3.2.4 Justificación del test a aplicar

```
##
##  Welch Two Sample t-test
##
## data:  gpa_female and gpa_male
## t = 7.1125, df = 4087.9, p-value = 6.694e-13
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.1107219       Inf
## sample estimates:
## mean of x mean of y
##  2.733070  2.589029
```

El pvalor del test (8.522e× 10-13) es inferior al nivel de significación (0.05). Además el valor observado 7.0787 no se encuentra dentro de la zona de aceptación del hipótesis H_0. Por tanto, podemos rechazar la hipótesis nula a favor de la alternativa y podemos concluir que en promedio la nota media del semestre de las alumnas son mayor que la de los alumnos.

4.3.3 ¿Qué variables tienen correlación?

```
round(cor(gpa[,1:5]),2)
```

```
##          sat tothrs hsize hsrank colgpa
## sat     1.00   0.05  0.06  -0.18   0.41
## tothrs  0.05   1.00 -0.04  -0.06   0.04
## hsize   0.06  -0.04  1.00   0.61  -0.03
## hsrank -0.18  -0.06  0.61   1.00  -0.34
## colgpa  0.41   0.04 -0.03  -0.34   1.00
```

Vemos que existe cierta correlación entre la variable hsize y hsrank.

4.4.4 Modelo de regresión lineal

```
# Estimacion del modelo lineal
Model <- lm(colgpa~sat+hsize+hsrank, data=gpa)
summary(Model)
```

```
##
## Call:
## lm(formula = colgpa ~ sat + hsize + hsrank, data = gpa)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.63352 -0.35355  0.02873  0.39524  1.83991
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.074e+00  6.840e-02   15.70   <2e-16 ***
## sat          1.543e-03  6.576e-05   23.46   <2e-16 ***
## hsize        7.208e-02  6.543e-03   11.02   <2e-16 ***
## hsrank      -4.006e-03  1.781e-04  -22.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.566 on 4133 degrees of freedom
## Multiple R-squared:  0.2607, Adjusted R-squared:  0.2601
## F-statistic: 485.7 on 3 and 4133 DF,  p-value: < 2.2e-16
```

El valor del R^2 es 0.2607, un valor bastante bajo. Lo cual indica que el modelo generado tiene poca calidad.

Prediccion:

```
input = data.frame(sat = 920,hsize = 67,hsrank = 4)
predict(Model, input)
```

```
##        1
## 7.306839
```

---

5. Conclusiones

Para el desarrollo de esta practica, hemos empezado con un preprocesado de los datos, que incluye normalización y limpieza de las variables para manejar los casos de ceros o valores perdidos y valores extremos (outliers).

Para el caso de valores perdidos, se ha hecho uso de un método de imputación basado en la distancia de los N vecinos, de tal forma que no tengamos que eliminar registros del conjunto de datos inicial y a la vez nos permite tener datos de cierta calidad al imputarlos de esta forma.

Para el caso de los valores extremos, se ha observado que aunque se tratan de valores atípicos, no son valores anómalos. Por lo que se ha incluido los valores para el análisis.

En un paso posterior, se ha realizado tres tipos de pruebas estadísticas sobre un conjunto de datos que contiene diferentes variables relativas a las notas de los estudiantes universitarios.

Para cada una de ellas, hemos podido ver cuáles son los resultados y conclusiones que llegamos a partir de las pruebas estadísticas realizadas.

El análisis de contraste de hipótesis nos ha permitido responder a las preguntas formuladad inicialmente con una probabilidad de 95%.

Con el análsis de correlaciones hemos podido estudiar la correlación entre las variable.

Y se ha intentado generar un modelo de regresión lineal para predecir la nota media de un estudiante .

```
# Guardar de los datos limpios en .csv
write.csv(gpa, "gpa_clean.csv")
```