Tipología y ciclo de vida de los datos

Práctica 2: Limpieza y validación de los datos

Autores: César Fernández Domínguez, Isabel Fernández Esparza

Junio 2019

Contents

---

1 Solución

1.1 Descripción del dataset

Para la realización de esta práctica se ha seleccionado un conjunto de datos relacionado con resultados académicos de estudiantes de dos colegios de Portugal disponible en el repositorio de datos *kaggle*. En concreto este conjunto de datos se ha obtenido del enlace: https://www.kaggle.com/uciml/student-alcohol-consumption

Este dataset contiene información de estudiantes de matemáticas en edad de estudios secundarios. Se puede utilizar para analizar cómo afectan a los estudiantes de secundaria sus circunstancias personales a la hora de tener voluntad de continuar con estudios de mayor nivel. Estas circunstancias personales podemos entenderlas desde el punto de vista de: nivel de estudios de los padres, trabajo de los padres, si tienen pareja o no…

Por otro lado, también se podría hacer un análisis para estudiar la relación entre el número de suspensos de los estudiantes y el nivel de estudio de los padres, la distancia de los alumnos a los colegios, cómo influye el hecho de disponer de ayuda extraescolar en los resultados escolares, el tiempo de estudio semanal, el consumo de alcohol tanto diario como semanal, el número de ausencias…

Este conjunto de datos se presenta en dos ficheros distintos, en formato CSV: student-mad.csv (asignatura de matemáticas) y student-por.csv (asignatura de portugues), uno por cada asignatura.

El objetivo de esta práctica es limpiar los datos, unificarlos y poder estimar un modelo que pueda predecir el número de suspensos de un estudiante de matemáticas atendiendo a los factores anteriormente descritos. Teniendo en cuenta que en el juego de datos tenemos información de dos colegios diferentes podemos también intentar analizar si las predicciones realizadas están también sesgadas por el colegio al que pertenezcan los alumnos o por el sexo.

A continuación, se presenta una descripción de los atributos, para cada estudiante, contenidos en los dos ficheros:

1. school - colegio al que pertenece el alumno (binario: 'GP' - Gabriel Pereira o 'MS' - Mousinho da Silveira)
2. sex - sexo (binario: 'F' - mujer o 'M' - hombre)
3. age - edad (numérico: entre 15 y 22 años)
4. address - tipo de residencia (binario: 'U' - urbana o 'R' - rural)
5. famsize - tamaño de la familia (binario: 'LE3' - menor o igual a 3 o 'GT3' - mayor que 3)
6. Pstatus - padres separados o no (binario: 'T' - viven juntos o 'A' - separados)
7. Medu - nivel educativo de la madre (numérico: 0 - ninguno, 1 - educación primaria (4º grado), 2 – entre 5º a 9º grado, 3 – educación secundaria o 4 – educación superior)
8. Fedu - nivel educativo del padre (numérico: 0 - ninguno, 1 - educación primaria (4º grado), 2 – entre 5º a 9º grado, 3 – educación secundaria o 4 – educación superior)
9. Mjob - trabajo de la madre (nominal: 'teacher', 'health' care related, civil 'services' (p.e. administrativa o policía), 'at_home' o 'other')
10. Fjob - trabajo del padre (nominal: 'teacher', 'health' care related, civil 'services' (p.e. administrativa o policía), 'at_home' o 'other')
11. reason - razón para elegir la escuela (nominal: cerca de 'home', school 'reputation', 'course' preferencia o 'other')
12. guardian - guardian del estudiante (nominal: 'mother', 'father' o 'other')
13. traveltime - tiempo de viaje desde casa a la escuela (numérico: 1 - <15 min., 2 - 15 a 30 min., 3 - 30 min. a 1 hour, o 4 - >1 hora)
14. studytime - tiempo de estudio semanal (numérico: 1 - <2 horas, 2 - 2 a 5 horas, 3 - 5 a 10 horas, o 4 - >10 horas)
15. failures - número de asignaturas suspensas (numérico: n si 1<=n<3, en cualquier otro caso 4)
16. schoolsup - apoyo educativo adicional (binario: yes o no)
17. famsup - ayuda educativa de la familia (binario: yes o no)
18. paid - clases privadas de las asignaturas (Matemáticas o Portugues) (binario: yes o no)
19. activities - actividades extra-escolares (binario: yes o no)
20. nursery - asistió a la guardería (binario: yes o no)
21. higher - el alumno quiere realizar estudios superiores (binario: yes o no)
22. internet - el alumno tiene Internet en casa (binario: yes o no)
23. romantic - el alumno tiene pareja o no (binario: yes o no)
24. famrel - calidad de la relación familiar (numérico: desde 1 - muy mal a 5 - excelente)
25. freetime - tiempo libre después de la escuela (numérico: desde 1 - muy poco a 5 - mucho)
26. goout - el alumno sale con amigo (numérico: desde 1 - muy poco a 5 - mucho)
27. Dalc - consumo de alcohol diario (numérico: desde 1 - muy poco a 5 - mucho)
28. Walc - consumo de alcohol durante el fin de semana (numérico: desde 1 - muy poco a 5 - mucho)
29. health - estado de salud del alumno (numérico: desde 1 - muy mal a 5 - muy bueno)
30. absences - número de ausencias del alumno (numérico: desde 0 a 93)

Además de los siguientes calificaciones relacionadas con las asignaturas de matemáticas y portugues:

31. G1 - calificación primer trimestre (numérico: entre 0 a 20)
32. G2 - calificación segundo trimestre (numérico: entre 0 a 20)
33. G3 - calificación tercer trimestre (numérico: entre 0 a 20)

1.2 Importancia y objetivos de los análisis

Según vemos en la descripción del dataset, se nos plantea la necesidad de evaluar de que forma afectan las distintas condiciones de cada estudiante para el resultado final del curso, es decir, para la nota final obtenida. Para esto, en primer lugar, necesitaremos definir alguna nueva variable que tenga en cuenta la nota obtenida durante los tres trimestres que dura el curso, como veremos en el siguiente apartado.

Son muchos los análisis y preguntas que se podrían intentar responder a partir de estos datos. Sin embargo, en nuestro caso, nos vamos centrar en las siguientes preguntas:

- ¿ Cuales son las variables que influyen más en la calificación de los estudiantes ?
- ¿ Es posible predecir cual será la calificación final de un estudiante en función de los otros atributos ?
- ¿ Los alumnos que dedican más tiempo al estudio sacan mejores notas ?
- ¿ Aquellos alumnos que van a clases particulares o reciben ayuda por parte de sus padres sacan mejores notas ?
- En general, ¿las chicas son mejores estudiantes que los chicos?

Todas estas preguntas vamos a intentar responderlas con nuestro siguiente análisis. Como podemos adivinar, este análisis puede resultar de vital importancia tanto para el profesorado y dirección de una escuela, como para los padres de estudiantes, para así conocer si las medidas tomadas dentro y fuera de la escuela llevan a unos mejores resultados académicos.

1.3 Integración y selección de los datos de interés a analizar.

Partimos de dos fichero CSV, student-mat.csv y student-por.csv, descargados del repositorio Kaggle.

```
sMat=read.table("data/student-mat.csv",sep=",",header=TRUE)
sPor=read.table("data/student-por.csv",sep=",",header=TRUE)

# Según el propietario de los datos, los alumnos que están presentes en ambas asignaturas
# pueden ser identificados por los siguientes atributos
sBoth=merge(sMat,sPor,by=c("school","sex","age","address","famsize","Pstatus","Medu",
                           "Fedu","Mjob","Fjob","reason","nursery","internet"))

# school, sex, age, address, famsize, Pstatus, Medu, Fedu, Mjob, Fjob, reason, nursery,
# internet, traveltime, studytime, failures, schoolsup, famsup, paid, activities, higher,
# romantic, famrel, freetime, goout, Dalc, Walc, health, absences, subject
```

Ambos ficheros de datos de estudiantes contienen 33 atributos (columnas). El fichero de estudiantes de la asignatura de portugués contiene 649 estudiantes y, el de la asignatura de matemáticas 395 estudiantes. Si mezclamos ambos ficheros, para obtener los alumnos que están en ambas asignaturas, obtenemos un total de 382 estudiantes.

Identificamos cada estudiante mediante los atributos indicados por el propietario del juego de datos. Generamos un identificador con la concatenación de estos atributos para cada estudiante. Después, en otro paso, convertiremos este identificador en un valor numérico que identifique a cada estudiante.

```
sMat$id = paste(sMat$school,sMat$sex,sMat$age,sMat$address,sMat$famsize,sMat$Pstatus,
                sMat$Medu,sMat$Fedu,sMat$Mjob,sMat$Fjob,sMat$reason,sMat$nursery,
                sMat$internet, sep="-")
```

```
sPor$id = paste(sPor$school,sPor$sex,sPor$age,sPor$address,sPor$famsize,sPor$Pstatus,
                sPor$Medu,sPor$Fedu,sPor$Mjob,sPor$Fjob,sPor$reason,sPor$nursery,
                sPor$internet, sep="-")
```

Creamos también una variable "score" que contendrá la media de las tres notas de los tres trimestres para cada alumno y asignatura. Luego, a partir de esta variable, creamos una variable categórica que exprese si un alumno ha aprobado o suspendido la asignatura. Además, creamos una nueva variable categórica que expresará la nota obtenida según la clasificación de cinco niveles utilizada en los grados Erasmus.

```
sMat$score = rowMeans(subset(sMat, select = c(G1, G2, G3)), na.rm = TRUE)
sMat$mark<-sMat$score
sMat$mark[sMat$score<10] <- "fail"
sMat$mark[sMat$score>=10] <- "pass"
sMat$mark <- as.factor(sMat$mark)
sMat$calification <- sMat$score
sMat$calification[(sMat$score<=20) & (sMat$score>=16)] <- "A"
sMat$calification[(sMat$score<16) & (sMat$score>=14)] <- "B"
sMat$calification[sMat$score<14 & sMat$score>=12] <- "C"
sMat$calification[sMat$score<12 & sMat$score>=10] <- "D"
sMat$calification[sMat$score<10] <- "F"
sMat$calification <- as.factor(sMat$calification)

sPor$score = rowMeans(subset(sPor, select = c(G1, G2, G3)), na.rm = TRUE)
sPor$mark<-sPor$score
sPor$mark[sPor$score<10] <- "fail"
sPor$mark[sPor$score>=10] <- "pass"
sPor$mark <- as.factor(sPor$mark)
sPor$calification <- sPor$score
sPor$calification[(sPor$score<=20) & (sPor$score>=16)] <- "A"
sPor$calification[(sPor$score<16) & (sPor$score>=14)] <- "B"
sPor$calification[sPor$score<14 & sPor$score>=12] <- "C"
sPor$calification[sPor$score<12 & sPor$score>=10] <- "D"
sPor$calification[sPor$score<10] <- "F"
sPor$calification <- as.factor(sPor$calification)
```

Creamos una nueva variable para identificar la asignatura asociada a cada instancia de nuestro dataset:

```
sMat$subject = 'Math'
sPor$subject = 'Portuguese'
students = rbind(sMat,sPor)
students$subject = as.factor(students$subject)
```

Ahora, a partir del identificador que anteriormente habiamos creado para cada estudiante, lo transformamos en un identificador numérico simple.

```
students = transform(students, id=as.numeric(factor(id)))
students$id = as.factor(students$id)
```

Tras estas transformaciones obtenemos un dataset con 1044 instancias y 38 atributos para un total de 662 estudiantes de ambas asignaturas.

Así, nuestro dataset nos queda como:

```
# Resumen
glimpse(students)
```

```
## Observations: 1,044
```

```
## Variables: 38
## $ school       <fct> GP, GP, GP, GP, GP, GP, GP, GP, GP, GP, GP, GP, G...
## $ sex          <fct> F, F, F, F, F, M, M, F, M, M, F, F, M, M, M, F, F...
## $ age          <int> 18, 17, 15, 15, 16, 16, 16, 17, 15, 15, 15, 15, 1...
## $ address      <fct> U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U...
## $ famsize      <fct> GT3, GT3, LE3, GT3, GT3, LE3, LE3, GT3, LE3, GT3,...
## $ Pstatus      <fct> A, T, T, T, T, T, T, A, A, T, T, T, T, T, A, T, T...
## $ Medu         <int> 4, 1, 1, 4, 3, 4, 2, 4, 3, 3, 4, 2, 4, 4, 2, 4, 4...
## $ Fedu         <int> 4, 1, 1, 2, 3, 3, 2, 4, 2, 4, 4, 1, 4, 3, 2, 4, 4...
## $ Mjob         <fct> at_home, at_home, at_home, health, other, service...
## $ Fjob         <fct> teacher, other, other, services, other, other, ot...
## $ reason       <fct> course, course, other, home, home, reputation, ho...
## $ guardian     <fct> mother, father, mother, mother, father, mother, m...
## $ traveltime   <int> 2, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 3, 1, 2, 1, 1, 1...
## $ studytime    <int> 2, 2, 2, 3, 2, 2, 2, 2, 2, 2, 2, 3, 1, 2, 3, 1, 3...
## $ failures     <int> 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ schoolsup    <fct> yes, no, yes, no, no, no, no, yes, no, no, no, no...
## $ famsup       <fct> no, yes, no, yes, yes, yes, no, yes, yes, yes, ye...
## $ paid         <fct> no, no, yes, yes, yes, yes, no, no, yes, yes, yes...
## $ activities   <fct> no, no, no, yes, no, yes, no, no, no, yes, no, ye...
## $ nursery      <fct> yes, no, yes, yes, yes, yes, yes, yes, yes, yes, ...
## $ higher       <fct> yes, yes, yes, yes, yes, yes, yes, yes, yes, yes,...
## $ internet     <fct> no, yes, yes, yes, no, yes, yes, no, yes, yes, ye...
## $ romantic     <fct> no, no, no, yes, no, no, no, no, no, no, no, no, ...
## $ famrel       <int> 4, 5, 4, 3, 4, 5, 4, 4, 4, 5, 3, 5, 4, 5, 4, 4, 3...
## $ freetime     <int> 3, 3, 3, 2, 3, 4, 4, 1, 2, 5, 3, 2, 3, 4, 5, 4, 2...
## $ goout        <int> 4, 3, 2, 2, 2, 2, 4, 4, 2, 1, 3, 2, 3, 3, 2, 4, 3...
## $ Dalc         <int> 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ Walc         <int> 1, 1, 3, 1, 2, 2, 1, 1, 1, 1, 2, 1, 3, 2, 1, 2, 2...
## $ health       <int> 3, 3, 3, 5, 5, 5, 3, 1, 1, 5, 2, 4, 5, 3, 3, 2, 2...
## $ absences     <int> 6, 4, 10, 2, 4, 10, 0, 6, 0, 0, 0, 4, 2, 2, 0, 4,...
## $ G1           <int> 5, 5, 7, 15, 6, 15, 12, 6, 16, 14, 10, 10, 14, 10...
## $ G2           <int> 6, 5, 8, 14, 10, 15, 12, 5, 18, 15, 8, 12, 14, 10...
## $ G3           <int> 6, 6, 10, 15, 10, 15, 11, 6, 19, 15, 9, 12, 14, 1...
## $ id           <fct> 184, 123, 39, 25, 72, 341, 336, 121, 279, 260, 32...
## $ score        <dbl> 5.666667, 5.333333, 8.333333, 14.666667, 8.666667...
## $ mark         <fct> fail, fail, fail, pass, fail, pass, pass, fail, p...
## $ calification <fct> F, F, F, B, F, B, D, F, A, B, F, D, B, D, B, B, C...
## $ subject      <fct> Math, Math, Math, Math, Math, Math, Math, Math, M...
```

Tenemos un total de 17 variables numéricas y 21 categóricas.

A continuación, mostramos las estadísticas básicas para cada variable de nuestro dataset final:

```
# Estadísticas básicas
summary(students)
```

```
##  school   sex          age        address famsize   Pstatus
##  GP:772   F:591   Min.   :15.00   R:285   GT3:738   A:121
##  MS:272   M:453   1st Qu.:16.00   U:759   LE3:306   T:923
##                   Median :17.00
##                   Mean   :16.73
##                   3rd Qu.:18.00
##                   Max.   :22.00
##
##       Medu            Fedu             Mjob           Fjob
```

```
##  Min.   :0.000   Min.   :0.000   at_home :194   at_home : 62
##  1st Qu.:2.000   1st Qu.:1.000   health  : 82   health  : 41
##  Median :3.000   Median :2.000   other   :399   other   :584
##  Mean   :2.603   Mean   :2.388   services:239   services:292
##  3rd Qu.:4.000   3rd Qu.:3.000   teacher :130   teacher : 65
##  Max.   :4.000   Max.   :4.000
##
##         reason      guardian     traveltime      studytime
##  course     :430   father:243   Min.   :1.000   Min.   :1.00
##  home       :258   mother:728   1st Qu.:1.000   1st Qu.:1.00
##  other      :108   other : 73   Median :1.000   Median :2.00
##  reputation:248                 Mean   :1.523   Mean   :1.97
##                                 3rd Qu.:2.000   3rd Qu.:2.00
##                                 Max.   :4.000   Max.   :4.00
##
##     failures      schoolsup famsup     paid     activities nursery
##  Min.   :0.0000   no :925   no :404   no :824   no :528    no :209
##  1st Qu.:0.0000   yes:119   yes:640   yes:220   yes:516    yes:835
##  Median :0.0000
##  Mean   :0.2644
##  3rd Qu.:0.0000
##  Max.   :3.0000
##
##  higher    internet  romantic      famrel         freetime
##  no : 89   no :217   no :673   Min.   :1.000   Min.   :1.000
##  yes:955   yes:827   yes:371   1st Qu.:4.000   1st Qu.:3.000
##                                Median :4.000   Median :3.000
##                                Mean   :3.936   Mean   :3.201
##                                3rd Qu.:5.000   3rd Qu.:4.000
##                                Max.   :5.000   Max.   :5.000
##
##      goout            Dalc            Walc           health
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:3.000
##  Median :3.000   Median :1.000   Median :2.000   Median :4.000
##  Mean   :3.156   Mean   :1.494   Mean   :2.284   Mean   :3.543
##  3rd Qu.:4.000   3rd Qu.:2.000   3rd Qu.:3.000   3rd Qu.:5.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000
##
##     absences            G1              G2              G3
##  Min.   : 0.000   Min.   : 0.00   Min.   : 0.00   Min.   : 0.00
##  1st Qu.: 0.000   1st Qu.: 9.00   1st Qu.: 9.00   1st Qu.:10.00
##  Median : 2.000   Median :11.00   Median :11.00   Median :11.00
##  Mean   : 4.435   Mean   :11.21   Mean   :11.25   Mean   :11.34
##  3rd Qu.: 6.000   3rd Qu.:13.00   3rd Qu.:13.00   3rd Qu.:14.00
##  Max.   :75.000   Max.   :19.00   Max.   :19.00   Max.   :20.00
##
##        id           score          mark     calification       subject
##  317    :   4   Min.   : 1.333   fail:321   A: 76        Math      :395
##  326    :   4   1st Qu.: 9.333   pass:723   B:142        Portuguese:649
##  360    :   4   Median :11.333              C:239
##  408    :   4   Mean   :11.267              D:266
##  158    :   3   3rd Qu.:13.333              F:321
##  185    :   3   Max.   :19.333
```

```
##  (Other):1022
```

Vemos que tenemos muchos más alumnos del colegio Gabriel Pereira (772) que del Mousinho da Silveira (272). Un total de 723 aprobados frente a 321 suspensos.

1.4 Limpieza de los datos

1.4.1 ¿Los datos tienen contienen ceros o elementos vacíos? ¿Cómo gestionaríamos cada uno de estos casos?

Primero vamos a comprobar si nuestro juego de datos contiene valores nulos.

```
# Con datos nulos
colSums(is.na(students))
```

```
##       school          sex          age      address      famsize
##            0            0            0            0            0
##      Pstatus         Medu         Fedu         Mjob         Fjob
##            0            0            0            0            0
##       reason     guardian   traveltime    studytime     failures
##            0            0            0            0            0
##    schoolsup       famsup         paid   activities      nursery
##            0            0            0            0            0
##       higher     internet     romantic       famrel     freetime
##            0            0            0            0            0
##        goout         Dalc         Walc       health     absences
##            0            0            0            0            0
##           G1           G2           G3           id        score
##            0            0            0            0            0
##         mark calification      subject
##            0            0            0
```

vemos que no existen valores nulos.

Ahora vamos a comprobar si existen valores vacíos

```
# Con datos ""
colSums(students=="")
```

```
##       school          sex          age      address      famsize
##            0            0            0            0            0
##      Pstatus         Medu         Fedu         Mjob         Fjob
##            0            0            0            0            0
##       reason     guardian   traveltime    studytime     failures
##            0            0            0            0            0
##    schoolsup       famsup         paid   activities      nursery
##            0            0            0            0            0
##       higher     internet     romantic       famrel     freetime
##            0            0            0            0            0
##        goout         Dalc         Walc       health     absences
##            0            0            0            0            0
##           G1           G2           G3           id        score
##            0            0            0            0            0
##         mark calification      subject
##            0            0            0
```

Se observa que tampoco existen valores vacíos.

Por lo tanto, como no existen ni valores nulos ni elementos vacios en nuestro dataset no tenemos que hacer nada a este respecto.

Comprobamos, también, la existencia de valores cero.

```
colSums(students==0)
```

```
##       school          sex          age      address      famsize
##            0            0            0            0            0
##      Pstatus         Medu         Fedu         Mjob         Fjob
##            0            9            9            0            0
##       reason     guardian   traveltime    studytime     failures
##            0            0            0            0          861
##    schoolsup       famsup         paid   activities      nursery
##            0            0            0            0            0
##       higher     internet     romantic       famrel     freetime
##            0            0            0            0            0
##        goout         Dalc         Walc       health     absences
##            0            0            0            0          359
##           G1           G2           G3           id        score
##            1           20           53            0            0
##         mark calification      subject
##            0            0            0
```

En este caso, vemos que algunos atributos contienen valores cero. Sin embargo, estos valores corresponden a variables numéricas y comprobamos la validez los mismos.

Por último, visualizamos el número de valores únicos por cada atributo, comprobando su correspondencia con los datos.

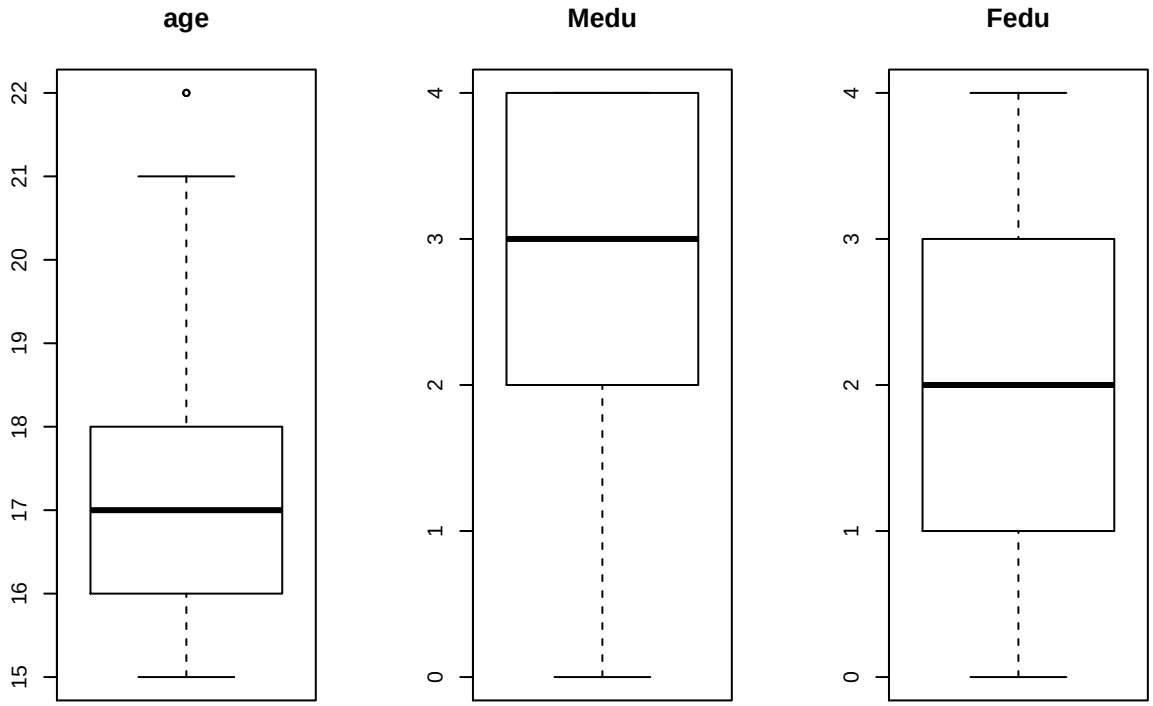
```
# Valores únicos
apply(students,2, function(x) length(unique(x)))
```

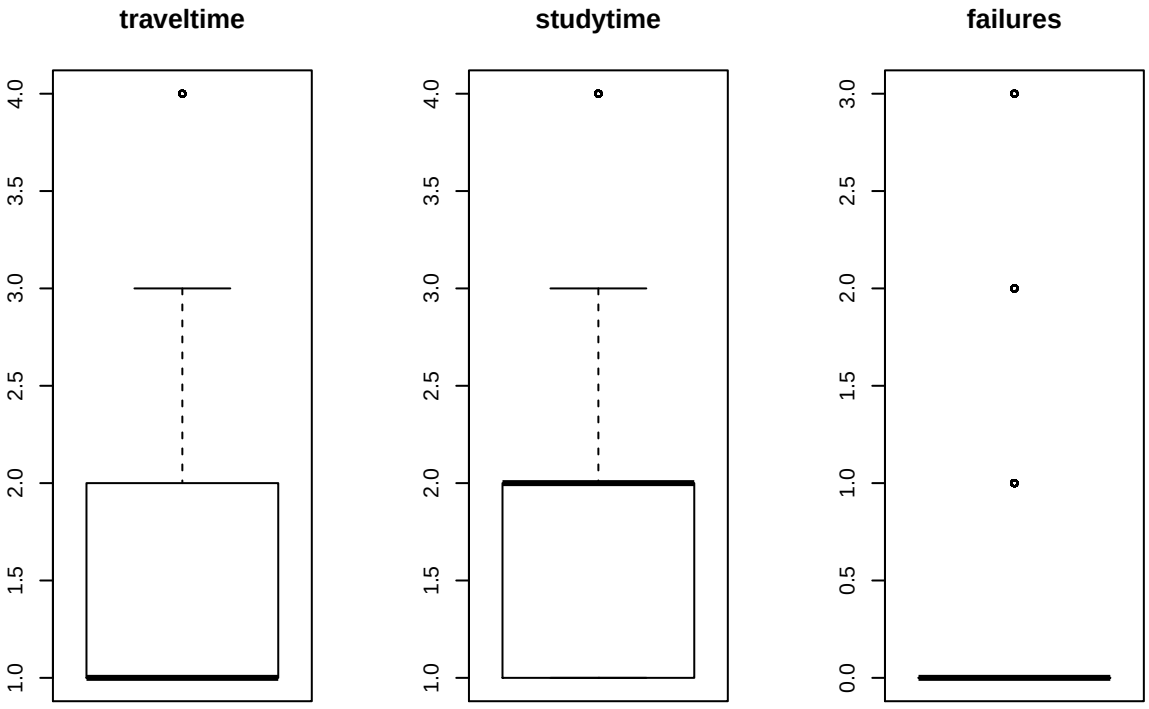
```
##       school          sex          age      address      famsize
##            2            2            8            2            2
##      Pstatus         Medu         Fedu         Mjob         Fjob
##            2            5            5            5            5
##       reason     guardian   traveltime    studytime     failures
##            4            3            4            4            4
##    schoolsup       famsup         paid   activities      nursery
##            2            2            2            2            2
##       higher     internet     romantic       famrel     freetime
##            2            2            2            5            5
##        goout         Dalc         Walc       health     absences
##            5            5            5            5           35
##           G1           G2           G3           id        score
##           18           17           19          662           54
##         mark calification      subject
##            2            5            2
```

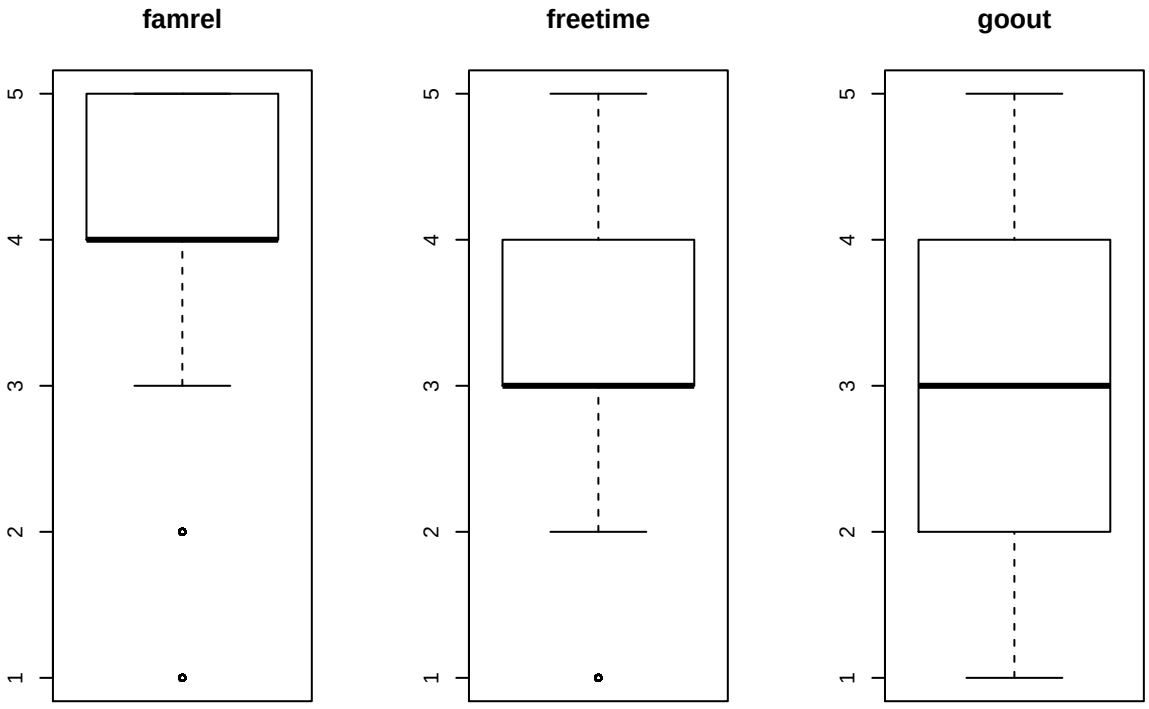
1.4.2 Identificación y tratamiento de valores extremos.

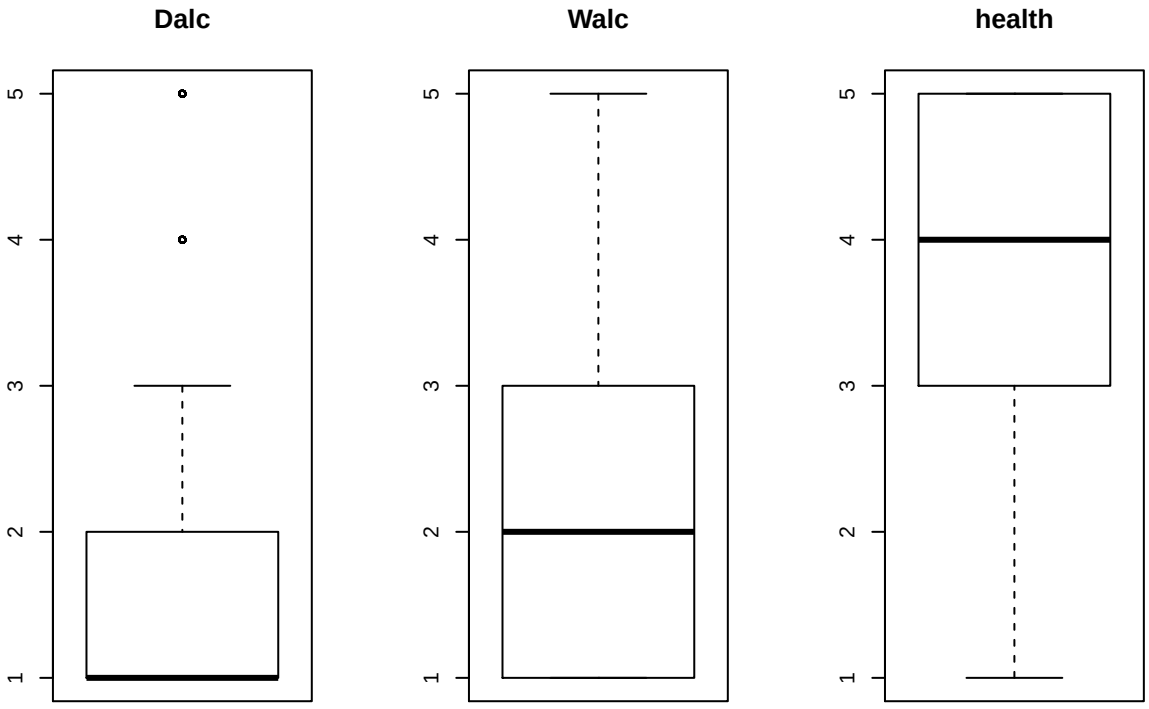
Los valores extremos o outliers son aquellas observaciones que están fuera de 1,5*IQR, donde IQR es la diferencia entre los cuartiles 75 y 25. Para buscar los outliers en nuestro juego de datos recorremos el dataset para encontrar todas aquellas variables numéricas y hacemos la representación gráfica de los outliers de cada una de ellas.

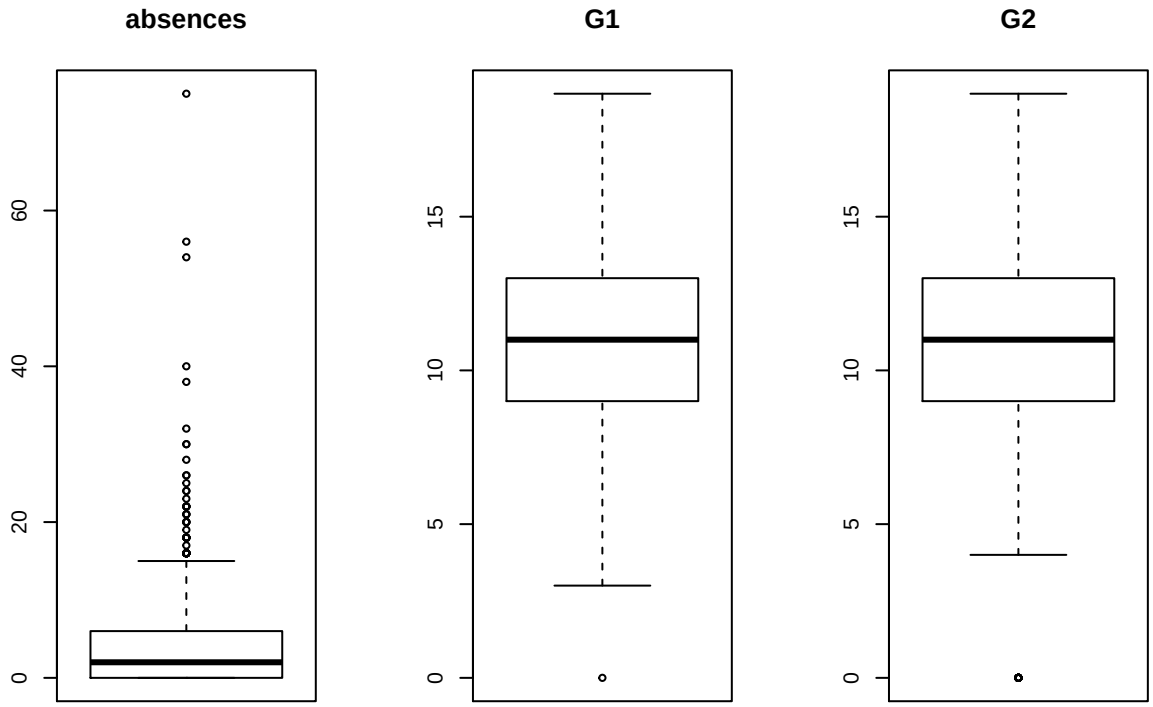
```
par(mfrow=c(1,3))
for(i in 1:ncol(students)) {
  if (is.numeric(students[,i])){
    boxplot(students[,i], main = colnames(students)[i], width = 100)
  }
}
```

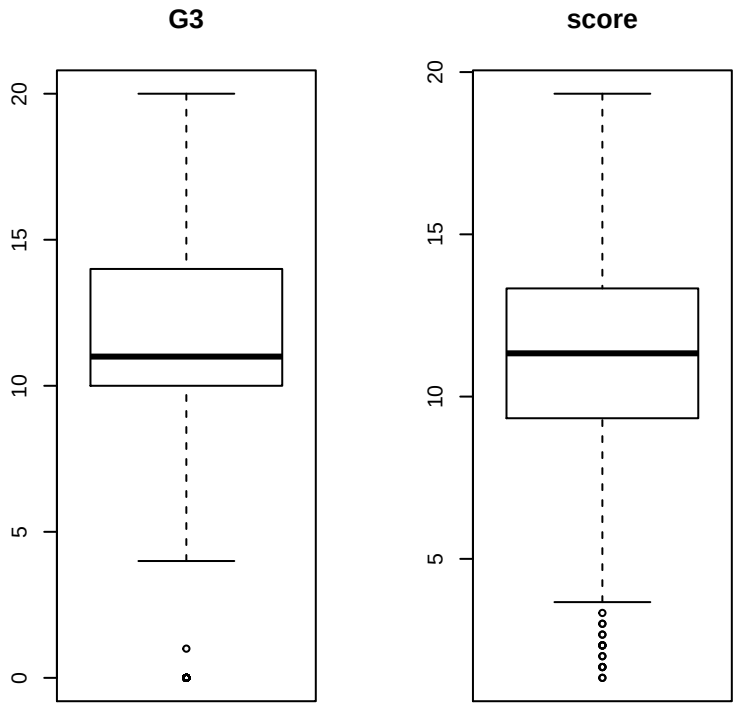



traveltime
studytime
failures
1.0
1.5
2.0
2.5
3.0
3.5
4.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0


famrel
freetime
goout
1
2
3
4
5


Dalc
Walc
health
1
2
3
4
5


absences
G1
G2
0
20
40
60
0
5
10
15
0
5
10
15



Vamos a analizar los diferentes outliers para cada una de las variables numéricas:

- **Variable Age**: Vemos que hay un outlier en el valor 22. En este caso consideramos que este valor no debería eliminarse y deberia tratarse como uno más. Probablemente el hecho de que haya alumnos de 22 años en el mismos curso que alumnos de 18 años estará relacionado con algunas de las variables que queremos analizar, principalmente las notas.
- **Variables Medu y Fedu**: No tienen valores extremos.
- **Variable traveltime**: Vemos que hay un outlier en el valor 4. Es decir, se dan casos extremos en los que los alumnos tardan 4 horas en llegar al colegio. Vamos a ver cuántas veces se da este valor en nuestra muestra

```
length(boxplot.stats(students[,c("traveltime")])$out)
```

```
## [1] 24
```

Vemos que tenemos 24 alumnos que tardan más de 3 horas en ir al colegio. El tiempo que tardan los alumnos en llegar al colegio no es una varaiable sobre la que queramos realizar hipótesis. Como además, el número total de outliers para este atributo es 24, esto representa el 2% de la muestra así que los eliminamos.

```
students = students[students$traveltime<=3,]
```

- **Variable studytime**: Vemos que, en este caso, también tiene un outlier en el valor 4. El tiempo de estudio de los alumnos sí que pensamos que es una variable que puede influir en los resultados finales y es una de los atributos que vamos a tener en cuenta en nuestros análisis. Sin embargo, no consideramos que el valor 4 sea un outlier que haya ni que eliminar ni tratar, puesto que es un valor aceptable para nuestro estudio. Luego con este outlier no hacemos nada.
- **Variable failures**: Lo mismo pasa con la variable que mide los suspensos. Los valores que aparecen

representados como outliers son precisamente los valores que pueden ser relevantes para nuestro estudio, luego no hacemos nada con ellos.

- **Variable freetime**: La variable freetime tiene un outlier en el valor 1, vamos a ver cuántos estudiantes cumplen esta condición.

```
nrow(students[students$freetime==1,])
```

```
## [1] 62
```

Son 62 alumnos que tienen muy poco tiempo libre entre semana. Puesto que este valor está relacionado con el tiempo de estudio semanal y con las ayudas extrasescolares que puedan recibir los alumnos decidimos mantenerlos también.

- **Variable famrel**: Tenemos outliers en los valores 1 y 2. Estos valores miden la calidad de las relaciones familiares. Entendiendo que estos valores se han obtenido a través de una encuesta realizada a los propios alumnos, el valor de 1 es tan bajo que pensamos que puede deberse a una manipulación por parte de los alumnos en las respuestas. Consideramos que una forma de tratar estos outliers es reemplazarlos por la moda (valor más repetido). En el caso de los valores de 2, los dejaremos como están puesto que no nos parece que puedan considerarse como outliers. Para calcular la moda utilizamos una tabla de frecuencia para contar el número de veces que se repite cada valor:

```
table(students$famrel)
```

```
##
##   1   2   3   4   5
##  28  47 168 501 276
```

Vemos que el valor más repetido es el 4, luego sustituimos aquellas columnas que tengan el valor 1 en la columna famrel por 4

```
students$famrel[students$famrel == 1] <- 4
```

- **Variables Dalc y Walc**: En el caso del consumo diario de alcohol, tenemos un outlier en los valores 4 y 5. Sin embargo, no tenemos estos outliers en el consumo del fin de semana. Pensamos, por tanto, que el hecho de que alumnos consuman mucho alcohol durante los días de diario puede estar relacionado con las notas que obtengan, así que vamos a dejar estos valores.
- **Variable absences**: Para el caso de las ausencias, nos interesa saber cuántos outliers tenemos y qué valores toman para poder analizar cómo tratarlos.

```
length(boxplot.stats(students[,c("absences")])$out)
```

```
## [1] 54
```

```
boxplot.stats(students[,c("absences")])$out
```

```
##  [1] 16 16 25 54 18 26 20 18 16 16 56 24 18 28 22 16 18 20 16 21 75 22 30
## [24] 19 20 38 18 20 22 40 23 16 17 16 16 24 22 16 32 16 16 30 21 16 18 16
## [47] 26 16 16 22 18 18 16 21
```

```
students[students$absences>=40,]
```

```
##     school sex age address famsize Pstatus Medu Fedu  Mjob     Fjob
## 75      GP   F  16       U     GT3       T    3    3 other services
## 184     GP   F  17       U     LE3       T    3    3 other    other
## 277     GP   F  18       R     GT3       A    3    2 other services
## 316     GP   F  19       R     GT3       T    2    3 other    other
##         reason guardian traveltime studytime failures schoolsup famsup
## 75        home   mother          1         2        0       yes    yes
## 184 reputation   mother          1         2        0        no    yes
```

```
## 277       home   mother          2         2        0        no     no
## 316 reputation    other          1         3        1        no     no
##     paid activities nursery higher internet romantic famrel freetime goout
## 75   yes        yes     yes    yes      yes       no      4        3     3
## 184   no        yes     yes    yes      yes      yes      5        3     3
## 277   no         no      no     no      yes      yes      4        1     1
## 316   no         no     yes    yes      yes      yes      4        1     2
##     Dalc Walc health absences G1 G2 G3  id     score mark calification
## 75     2    4      5       54 11 12 11  75 11.333333 pass            D
## 184    2    3      1       56  9  9  8 172  8.666667 fail            F
## 277    1    1      5       75 10  9  9 175  9.333333 fail            F
## 316    1    1      3       40 13 11 11 225 11.666667 pass            D
##     subject
## 75     Math
## 184    Math
## 277    Math
## 316    Math
```

A la vista de los resultados vemos que tenemos 54 outliers en las ausencias, y que los valores que toman en estos outliers son: 16 16 25 54 18 26 20 18 16 16 56 24 18 28 22 16 18 20 16 21 75 22 30 19 20 38 18 20 22 40 23 16 17 16 16 24 22 16 32 Nos parece que los valores que están en un intervalo de (10,30) son admisibles, sin embargo, vemos que hay valores que pasan de 40, que podrían eliminarse.

En total, vamos a contar cuántos datos tenemos para ausencias mayores o iguales que 40

```
nrow(students[students$absences>=40,])
```

```
## [1] 4
```

Vemos que tenemos 4 valores, luego decidimos eliminarlos de nuestro conjunto de datos.

```
students = students[students$absences<40,]
```

- **Variables G1, G2, G3 y score**: Las columnas relativas a las notas van a ser, principalmente, las que sean objeto de nuestro estudio. En este caso, tenemos outliers en notas que son perfectamente posibles, luego no vamos a eliminarlo puesto que pensamos que estos valores afectarán a los estudios e hipótesis que queramos analizar.

1.5 Análisis de los datos

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

En este apartado, vamos a realizar una selección de atributos que podrán ser de utilidad para los análisis posteriores.

Por ejemplo, podemos discriminar nuestros datos según su género: hombre o mujer (male o female):

```
# Agrupación por sexo de los estudiantes
students.male <- students[students$sex == "M",]
students.female <- students[students$sex == "F",]
```

Esta agrupación nos podrá servir para intentar responder a la pregunta de si las chicas son, en general, mejores estudiantes que los chicos.

Por otra parte, también podemos agrupar nuestros datos dependiendo de si reciben algún tipo de ayuda en el estudio, por ejemplo: mediante clases particulares, ayuda familiar o soporte extra en el colegio.

```
# Agrupación por si reciben clases particules pagadas o no
students.paid <- students[students$paid == "yes",]
```

```
students.nopaid <- students[students$paid == "no",]

# Agrupación por si reciben soporte por parte de la familia
students.famsup <- students[students$famsup == "yes",]
students.nofamsup <- students[students$famsup == "no",]

# Agrupación por si reciben ayuda extra escolar
students.schoolsup <- students[students$schoolsup == "yes",]
students.noschoolsup <- students[students$schoolsup == "no",]
```

También, vamos a realizar una agrupación de los estudiantes dependiendo del número de horas que dediquen semanalmente al estudio. En este caso, agrupamos por aquellos que dedican 3 o más horas semanalmente al estudio y aquellos que dedican menos de 3 horas semanales.

```
# Agrupación por tiempo dedicado al estudio
students.studytime <- students[students$studytime >= 3,]
students.nostudytime <- students[students$studytime < 3,]
```

Como vemos, resulta muy sencillo realizar la agrupación de nuestros datos dependiendo de distintas variables. Así, por ejemplo, también podremos disociar nuestros datos dependiendo del nivel de estudios de los padres.

```
# Agrupación por estudios de los padres
students.parentedu <- students[(students$Medu >= 3) | (students$Fedu >= 3),]
students.parentnoedu <- students[(students$Medu < 3) & (students$Fedu < 3),]
```

o, agrupando por edad de los estudiantes:

```
# Agrupación por edad
students.mayores <- students[students$age >= 16,]
students.menores <- students[students$age < 16,]
```

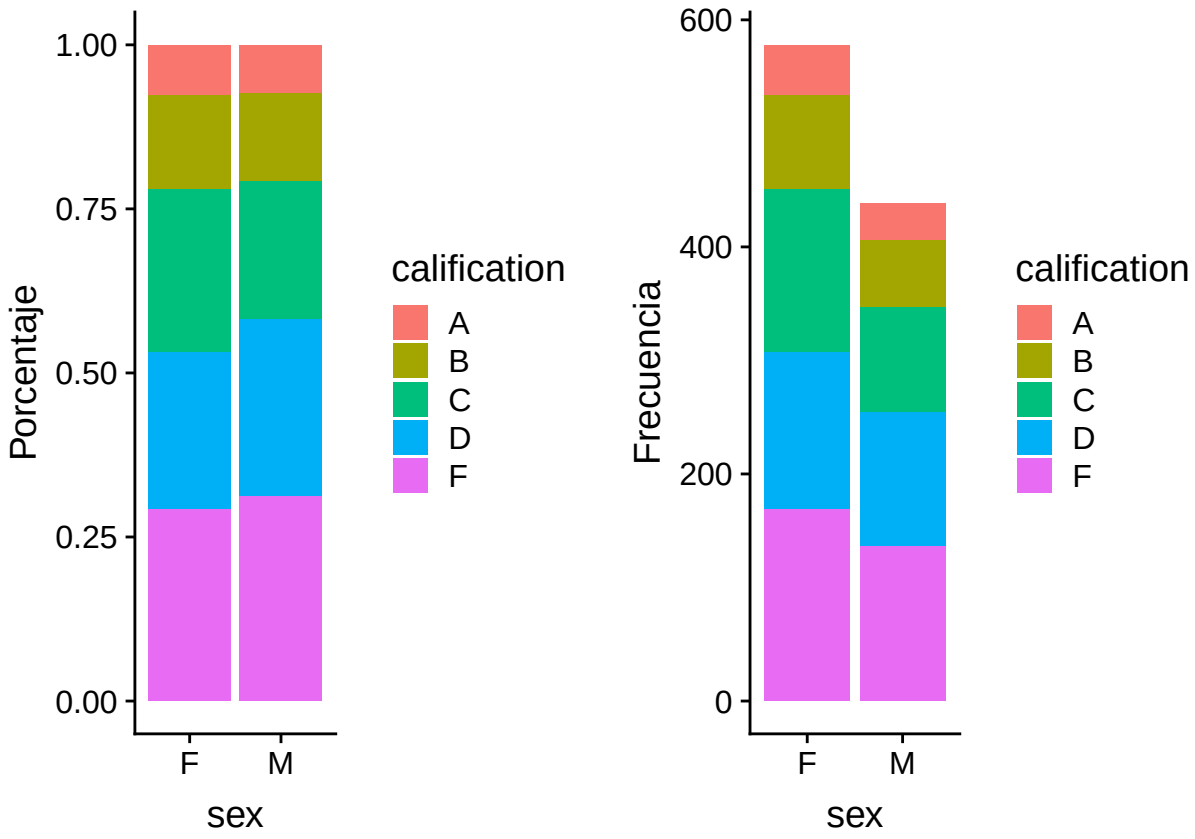
Estas agrupaciones de los datos de nuestro dataset original no serán todas utilizadas en los siguientes análisis. O, como veremos en la siguiente sección, esta agrupación también la podemos realizar de forma visual.

1.5.2 Análisis visual

En este apartado vamos a realizar un análisis gráfico básico de nuestro dataset respecto a las variables objeto de nuestro estudio.

En primer lugar, vemos cual es la distribución de notas de los estudiantes entre chicos y chicas.

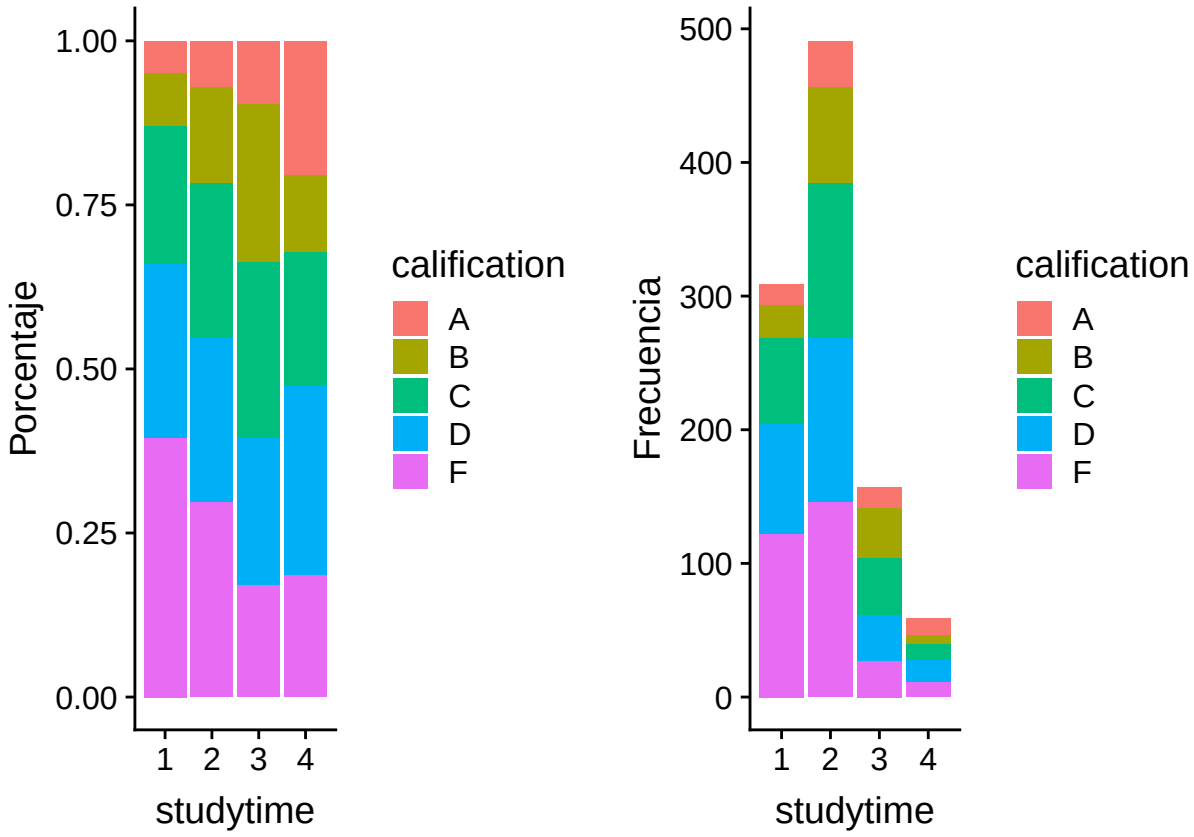
```
fig1 = ggplot(students,aes(x=sex,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")
fig2 = ggplot(students,aes(x=sex,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia")
grid.arrange(fig1, fig2, ncol=2)
```



Vemos como, en porcentaje, ambos grupos presentan notas similares. Quizá podemos apreciar un mayor porcentaje de aprobados en las chicas que en los chicos, pero nada significativo. También vemos que el número de muestras de chicas es superior al de chicos.

En las siguientes figuras, mostramos la calificación respecto a las horas de estudio semanales dedicadas por cada estudiante:

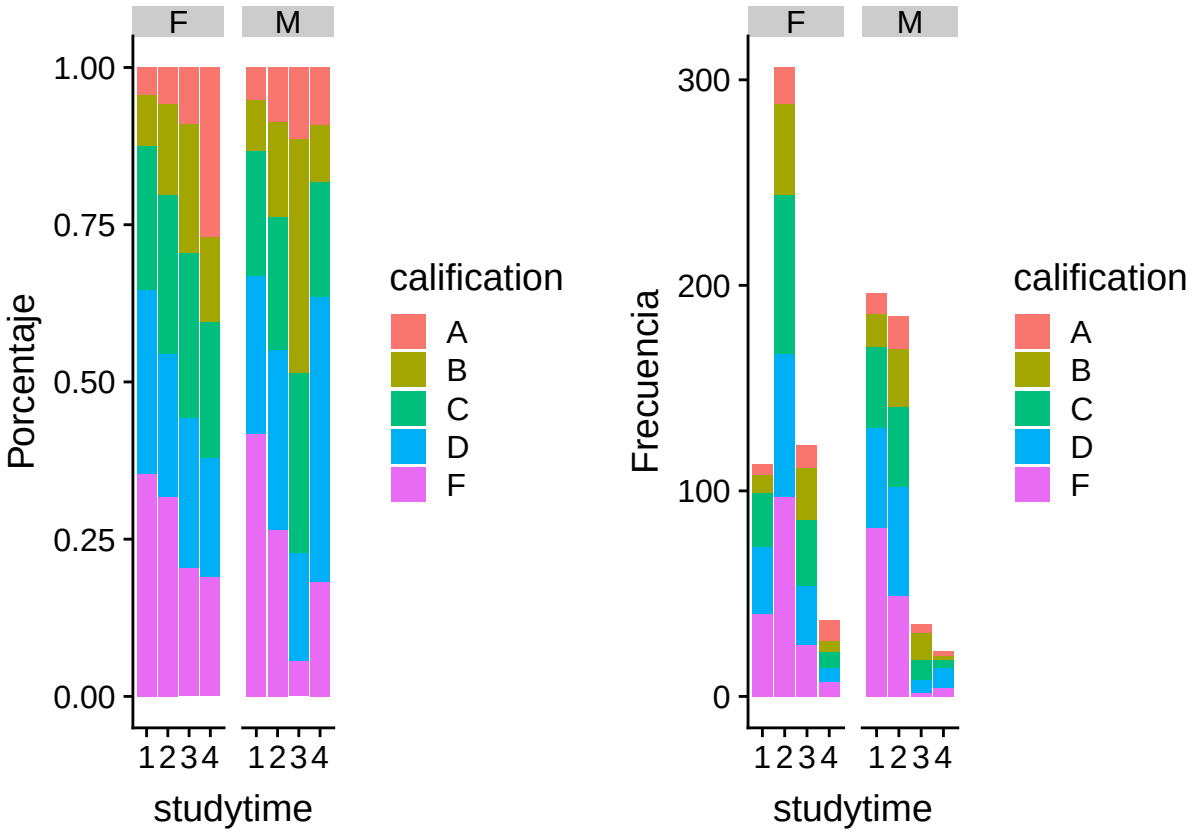
```
fig1 = ggplot(students,aes(x=studytime,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")
fig2 = ggplot(students,aes(x=studytime,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia")
grid.arrange(fig1, fig2, ncol=2)
```



Vemos como, a medida que se aumenta el número de horas semanales de estudio, aumenta también la probabilidad de obtener una nota más alta. Vemos como, a pesar de dedicar más de 4 horas semanales al estudio, el porcentaje de suspensos es mayor que el de aquellos que solamente dedican 3 horas semanales. Lo cual confirma lo que siempre se dice que con dedicar un poco de tiempo diariamente al estudio es suficiente para sacar buenas notas. También observamos que el número de alumnos que dedican más horas al estudio es mucho menor que aquellos que dedican más horas.

Si diferenciamos entre chicos y chicas:

```
fig1 = ggplot(students,aes(x=studytime,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")+facet_wrap(~sex )
fig2 = ggplot(students,aes(x=studytime,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia") +
  facet_wrap(~sex )
grid.arrange(fig1, fig2, ncol=2)
```

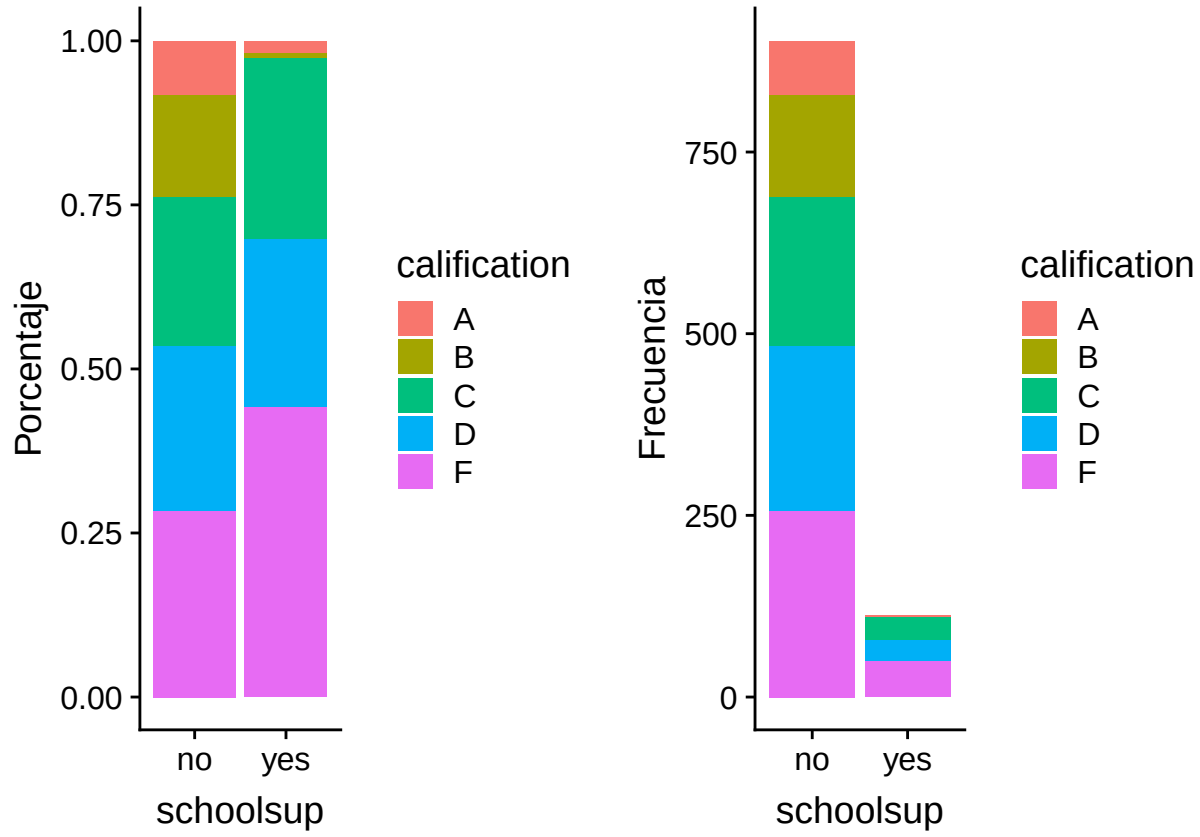


Vemos como en los chicos es significativo el porcentaje de suspensos en aquellos alumnos que dedican 4 horas o más semanales al estudio. Casi equiparable a aquellos que practicamente no dedican tiempo al estudio. En las chicas se ve una mayor progresión en mejoría de las notas a medida que se aumenta el tiempo de estudio. En las chicas, la gran mayoría afirma dedicar solamente 2 horas al estudio semanalmente.

Ahora vamos a visualizar cual es la influencia de la ayuda extra sobre las calificaciones de los estudiantes.

En primer lugar, visualizamos como influye el hecho de que los alumnos reciban ayuda extra por parte del colegio.

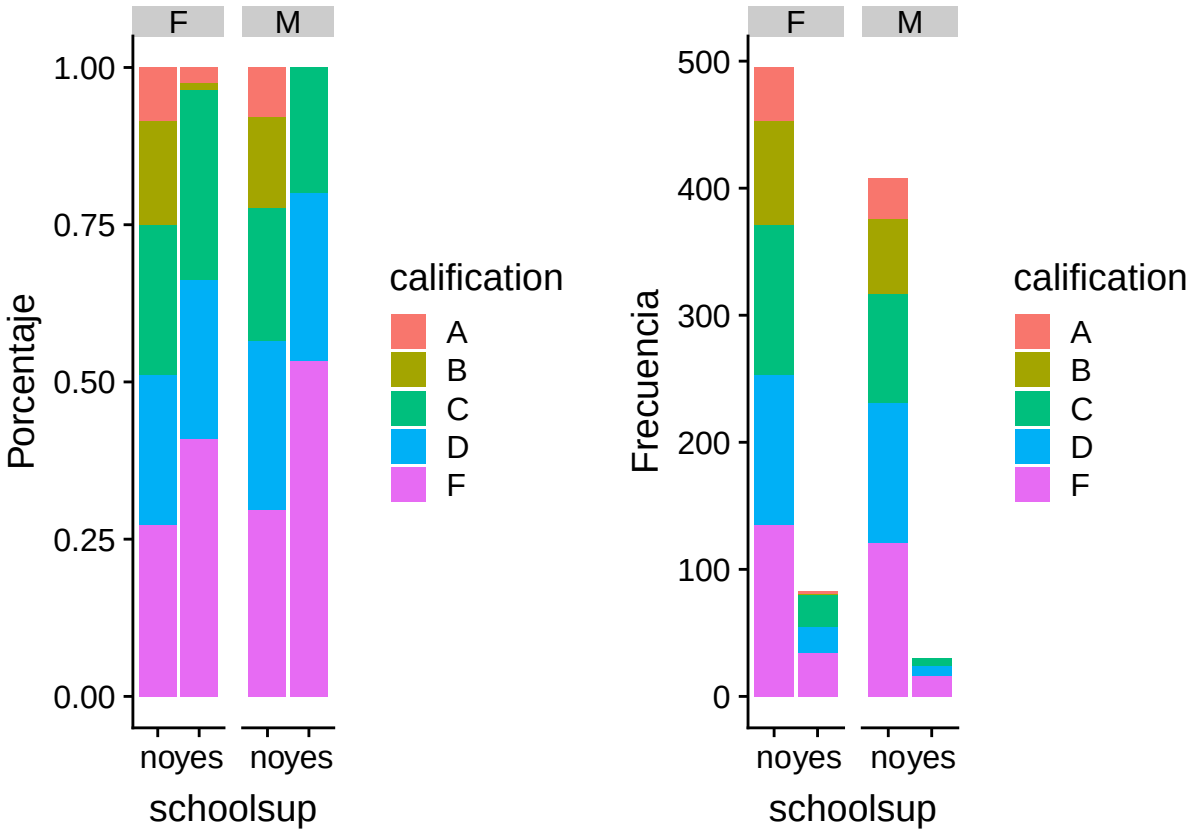
```
fig1 = ggplot(students,aes(x=schoolsup,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")
fig2 = ggplot(students,aes(x=schoolsup,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia")
grid.arrange(fig1, fig2, ncol=2)
```



En primer lugar, vemos que el número de alumnos que recibe ayuda extra es muy inferior a los que no la reciben. Seguramente esta ayuda solamente sea dada a aquellos alumnos que, por diversas razones, se vea que necesitan un soporte especial por parte del colegio (en general, malos estudiantes). Vemos que, a pesar de recibir ayuda extra en el colegio, practicamente el 70% de estos estudiantes suspende. Sí hay un pequeño porcentaje que aprueba con buena cualificación.

Diferenciando entre chicos y chicas:

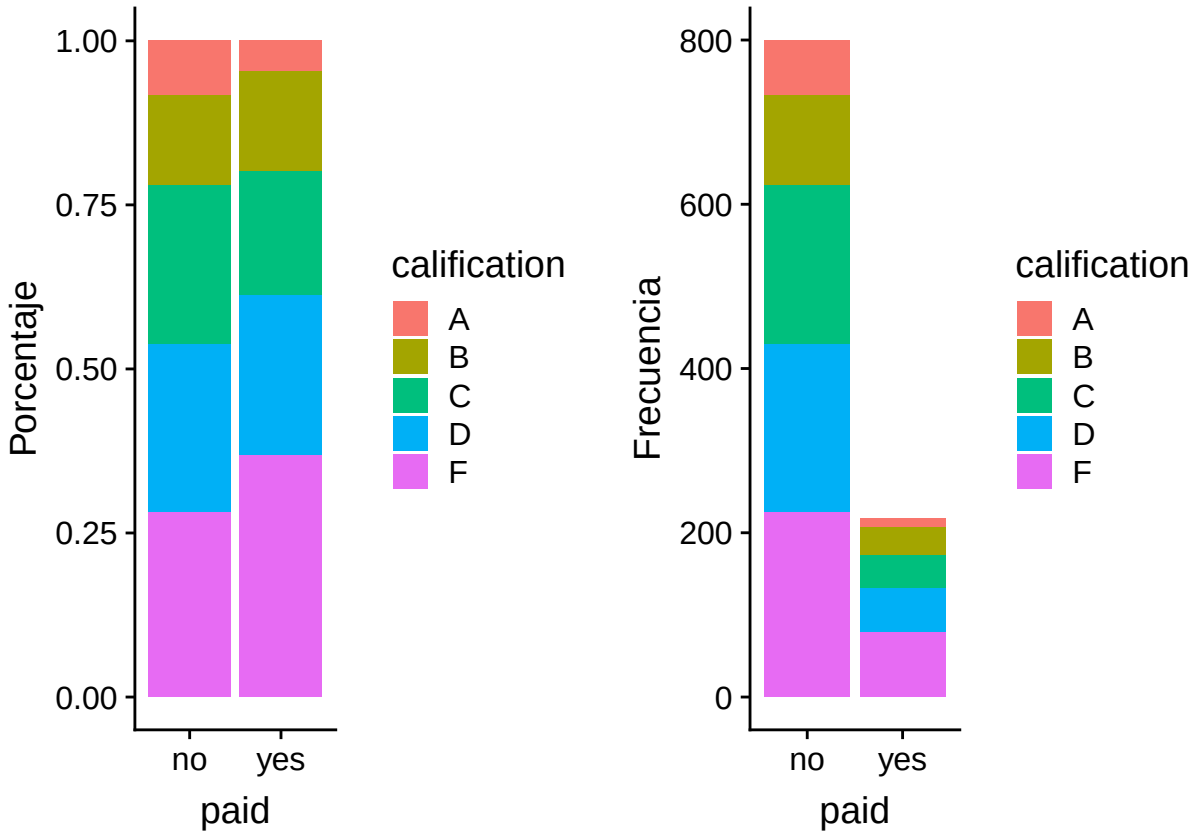
```
fig1 = ggplot(students,aes(x=schoolsup,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")+facet_wrap(~sex )
fig2 = ggplot(students,aes(x=schoolsup,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia") +
  facet_wrap(~sex )
grid.arrange(fig1, fig2, ncol=2)
```



Apreciamos que el hecho de recibir ayuda extra por parte del colegio, es mejor aprovechado por las chicas que por los chicos. En el caso de los chicos, unicamente aprueba alrededor del 20% de los chicos que recibe soporte extra por parte del colegio.

Analizando según vayan a clases particules o no, tenemos:

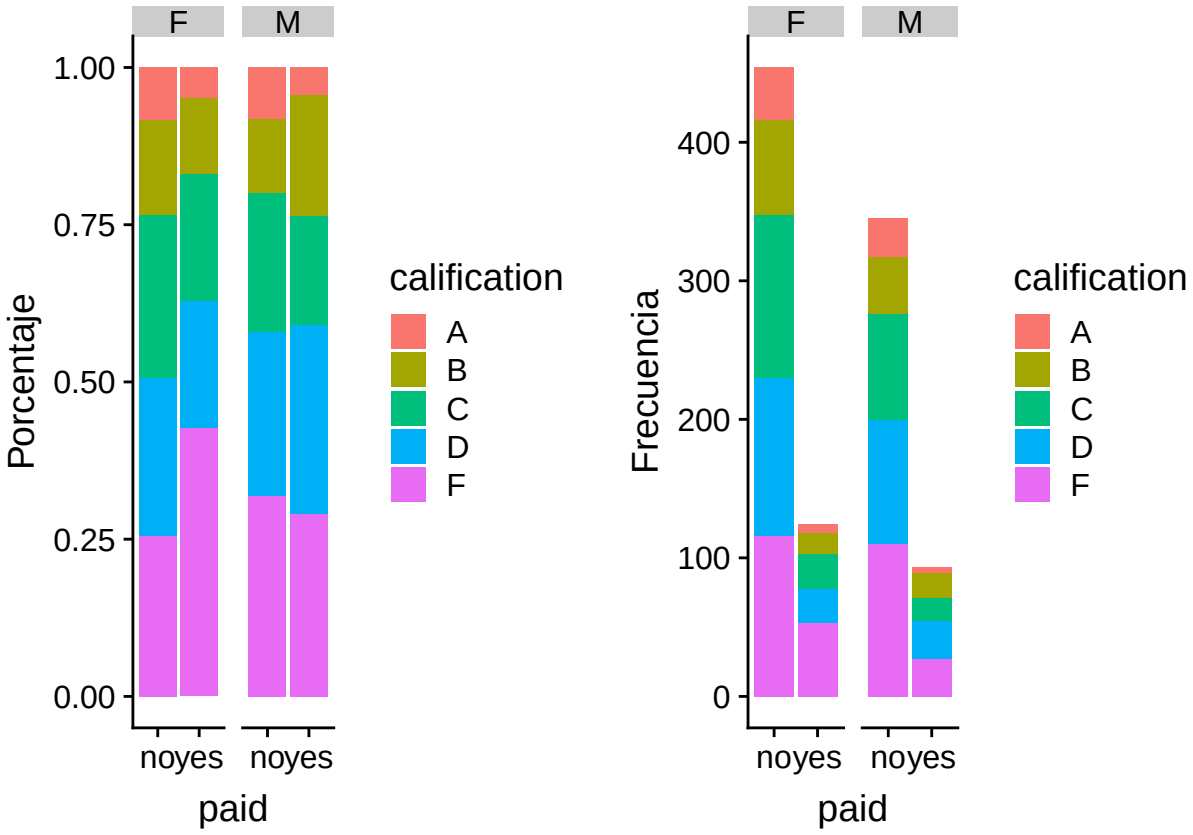
```
fig1 = ggplot(students,aes(x=paid,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")
fig2 = ggplot(students,aes(x=paid,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia")
grid.arrange(fig1, fig2, ncol=2)
```



En estas gráficas vemos como un porcentaje pequeño (alrededor del 20%) de los estudiantes recibe clases particulares. Vemos, también, que el porcentaje de aprobados es mayor en aquellos alumnos que van a clases particulares, pero no significativamente. Además es mayor el porcentaje de aprobados con nota más alta en aquellos alumnos que no reciben clases particulares. Posiblemente, porque el hecho de recibir clases particles se aplique más a estudiantes que necesitan un pequeño soporte extra.

De igual forma que en otros casos, si diferenciamos entre chicos y chicas, tenemos:

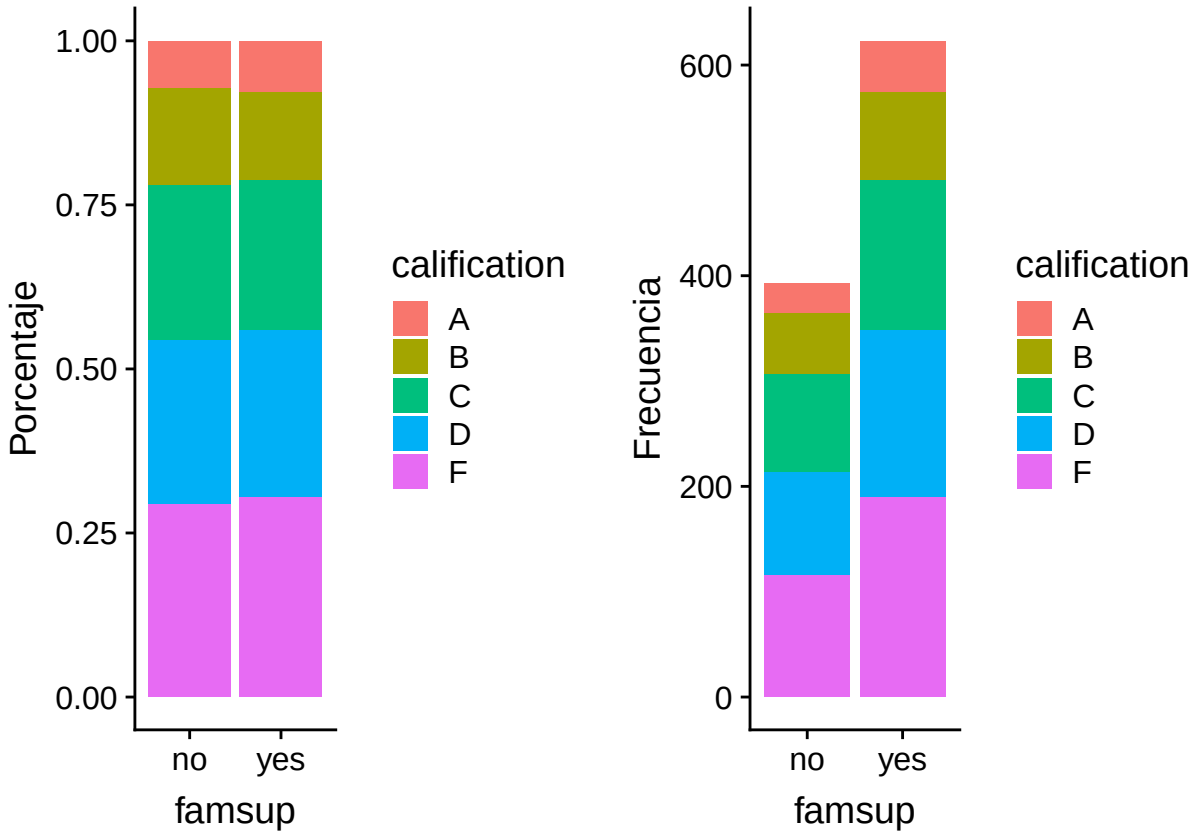
```
fig1 = ggplot(students,aes(x=paid,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")+facet_wrap(~sex )
fig2 = ggplot(students,aes(x=paid,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia") +
  facet_wrap(~sex )
grid.arrange(fig1, fig2, ncol=2)
```



En este caso no se aprecian diferencias significativas entre chicos y chicas. Quizá, podemos observar que, la mejora, en porcentaje, entre aquellos estudiantes que van a clases particulares y los que no, es mayor en las chicas.

Por último, visualizamos la distribución de las calificaciones de los estudiantes dependiendo de si reciben ayuda por parte de su familia o no:

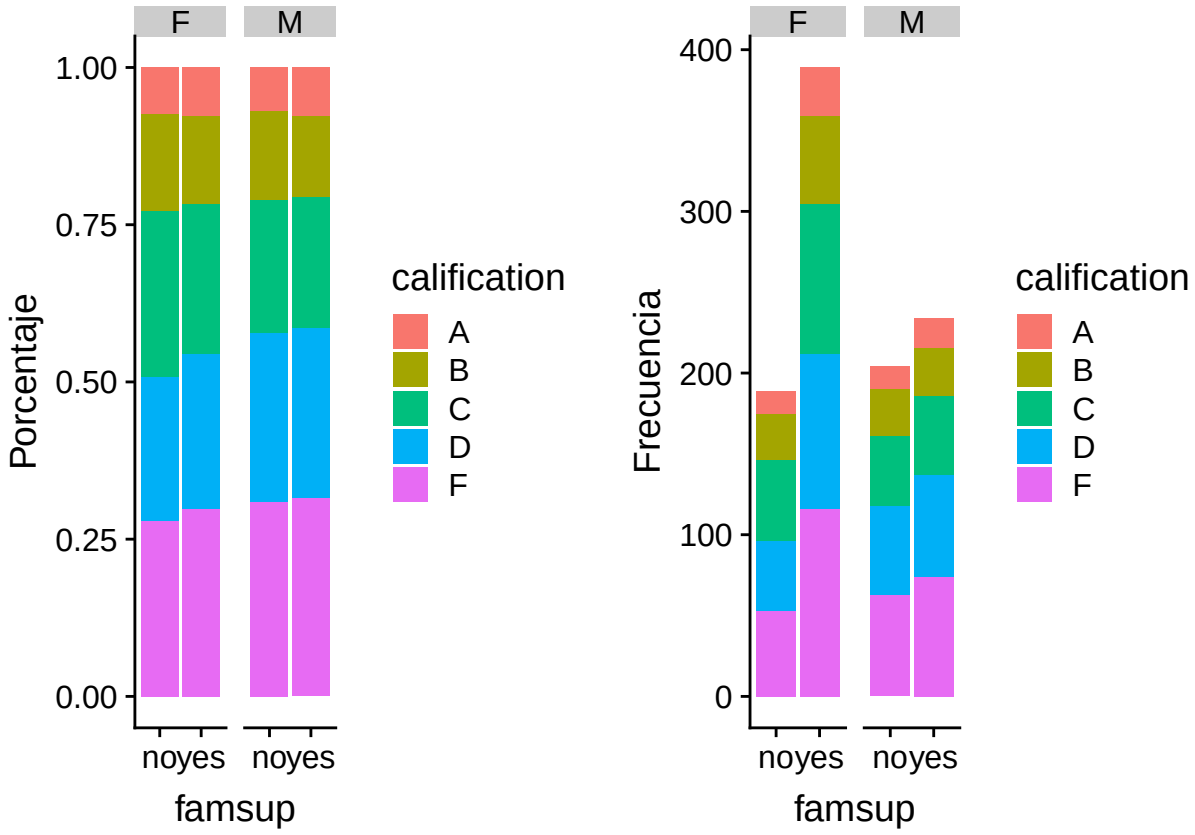
```
fig1 = ggplot(students,aes(x=famsup,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje")
fig2 = ggplot(students,aes(x=famsup,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia")
grid.arrange(fig1, fig2, ncol=2)
```



En este caso, es resaltable que el número de estudiantes que recibe ayuda de su familia es bastante mayor que los que no la reciben. Sin embargo, los resultados académicos son equiparables (gráfica de porcentajes).

Diferenciando entre chicos y chicas:

```
fig1 = ggplot(students,aes(x=famsup,fill=calification)) +
  geom_bar(position="fill") +
  ylab("Porcentaje") +
  facet_wrap(~sex )
fig2 = ggplot(students,aes(x=famsup,fill=calification)) +
  geom_bar() +
  ylab("Frecuencia") +
  facet_wrap(~sex )
grid.arrange(fig1, fig2, ncol=2)
```



Son casi el doble las chicas que reciben ayuda de sus familiares de las que no. En los chicos está más equilibrado. En cuando a los resultados académicos, vemos una ligera mejoría, ligerísima, entre las chicas que reciben ayuda y las que no.

1.5.3 Comprobación de la normalidad y homogeneidad de la varianza

Para la comprobación de que los valores que toman nuestras variables cuantitativas provienen de una población distribuida normalmente, aplicaremos distintas pruebas de normalidad, como son: el test de Anderson-Darling, el test de Kolmogorov-Smirnov, el test de Shapiro y el test de Cramer-von Mises.

Así, para cada uno de los tests, se comprueba que el valor que se obtiene de p. Si este valor es superior a nivel de significación prefijado $= 0{,}05$, se considera que la variable en cuestión sigue una distribución normal.

Definimos una función para aplicar facilmente los cuatro test de normalidad sobre cada variable numérica:

```
# Tests de normalidad
library(nortest)

# Función que aplica distintos test de normalidad sobre los datos de entrada
normTest <- function(data, name, alpha = 0.05) {
  # Anderson-Darling test
  ad_val = (ad.test(data)$p.value > alpha)
  # Kolmogorov-Smirnov test
  ks_val = (ks.test(data, pnorm, mean(data), sd(data))$p.value > alpha)
  # Shapiro test
  sh_val = (shapiro.test(data)$p.value > alpha)
  # Cramer-von Mises test
  csv_val = (cvm.test(data)$p.value > alpha)
```

```
  return (c(name,ad_val,ks_val,sh_val,csv_val))
}
```

El siguiente código ejecuta los test de normalidad para cada una de las variables numéricas de nuestro dataset:

```
col.names = colnames(students)
tmat <- matrix(NA, nrow = 0, ncol = 5)
colnames(tmat) <- c("Attribute","Anderson-Darling","Kolmogorov-Smirnov",
                    "Shapiro","Cramer-von Mises")
for (i in 1:ncol(students)) {
  if (is.integer(students[,i]) | is.numeric(students[,i])) {
    normTest(students[,i], col.names[i])
    tmat <- rbind(tmat, normTest(students[,i], col.names[i]))
  }
}
pander::pander(data.frame(tmat), split.table = 180)
```

Attribute	Anderson.Darling	Kolmogorov.Smirnov	Shapiro	Cramer.von.Mises
age	FALSE	FALSE	FALSE	FALSE
Medu	FALSE	FALSE	FALSE	FALSE
Fedu	FALSE	FALSE	FALSE	FALSE
traveltime	FALSE	FALSE	FALSE	FALSE
studytime	FALSE	FALSE	FALSE	FALSE
failures	FALSE	FALSE	FALSE	FALSE
famrel	FALSE	FALSE	FALSE	FALSE
freetime	FALSE	FALSE	FALSE	FALSE
goout	FALSE	FALSE	FALSE	FALSE
Dalc	FALSE	FALSE	FALSE	FALSE
Walc	FALSE	FALSE	FALSE	FALSE
health	FALSE	FALSE	FALSE	FALSE
absences	FALSE	FALSE	FALSE	FALSE
G1	FALSE	FALSE	FALSE	FALSE
G2	FALSE	FALSE	FALSE	FALSE
G3	FALSE	FALSE	FALSE	FALSE
score	FALSE	FALSE	FALSE	FALSE

Luego, a la vista de los resultados obtenidos en los diferentes tests de normalidad, vemos que ninguna de las variables numéricas que tenemos en nuestro juego de datos sigue una distribución normal con respecto al conjunto total de los datos.

Ahora vamos a ver si estas variables numéricas siguen una distribución normal en cada uno de los grupos que hemos separado para realizar nuestro análisis. Es decir, vamos a ver si:

- ¿seguirán las notas una distribución normal en el conjunto de hombres o de mujeres?
- ¿Y en el caso de que el tiempo de estudio sea de más de 3 horas o de menos?
- ¿Y de los alumnos mayores de 16 años o menores?
- ¿Y si separamos los datos entre alumnos que reciben clases particulares pagadas y no?
- ¿Y si reciben ayuda extraescolar de familia o en el colegio?

```
tmat <- matrix(NA, nrow = 0, ncol = 5)
colnames(tmat) <- c("Attribute","Anderson-Darling","Kolmogorov-Smirnov",
                    "Shapiro","Cramer-von Mises")
tmat <- rbind(tmat, normTest(students.female[,c("score")],"score~female"))
```

```
tmat <- rbind(tmat, normTest(students.male[,c("score")],"score~male"))
tmat <- rbind(tmat, normTest(students.studytime[,c("score")],"score~studytime"))
tmat <- rbind(tmat, normTest(students.nostudytime[,c("score")],"score~nostudytime"))
tmat <- rbind(tmat, normTest(students.mayores[,c("score")], "score~mayores"))
tmat <- rbind(tmat, normTest(students.menores[,c("score")], "score~menores"))
tmat <- rbind(tmat, normTest(students.paid[,c("score")], "score~paid"))
tmat <- rbind(tmat, normTest(students.nopaid[,c("score")], "score~nopaid"))
tmat <- rbind(tmat, normTest(students.schoolsup[,c("score")], "score~schoolsup"))
tmat <- rbind(tmat, normTest(students.noschoolsup[,c("score")], "score~noschoolsup"))
tmat <- rbind(tmat, normTest(students.famsup[,c("score")], "score~famsup"))
tmat <- rbind(tmat, normTest(students.nofamsup[,c("score")], "score~nofamsup"))
pander::pander(data.frame(tmat), split.table = 180)
```

Attribute	Anderson.Darling	Kolmogorov.Smirnov	Shapiro	Cramer.von.Mises
score~female	FALSE	TRUE	FALSE	FALSE
score~male	FALSE	TRUE	FALSE	FALSE
score~studytime	TRUE	TRUE	FALSE	TRUE
score~nostudytime	FALSE	FALSE	FALSE	FALSE
score~mayores	FALSE	TRUE	FALSE	FALSE
score~menores	TRUE	TRUE	TRUE	TRUE
score~paid	TRUE	TRUE	TRUE	TRUE
score~nopaid	FALSE	FALSE	FALSE	FALSE
score~schoolsup	TRUE	TRUE	TRUE	TRUE
score~noschoolsup	FALSE	FALSE	FALSE	FALSE
score~famsup	FALSE	TRUE	FALSE	TRUE
score~nofamsup	FALSE	TRUE	FALSE	FALSE

Según los resultados obtenidos vemos que, para los alumnos menores de 16 años, las notas medias sí siguen una distribución normal, y para el caso de los que reciben clases particulares pagadas también, así como, para aquellos que reciben apoyo educativo adicional . Para el caso de las alumnas femeninas, y de los grupos que reciben o no ayuda extraescolar o familiar, el test de Kolmogorov sí indica que las variables de notas medias siguen una distribución normal, pero el resto de test indican lo contrario.

Vamos a estudiar ahora la homogeneidad de la varianza para los diferentes grupos que queremos analizar. En primer lugar vamos a analizar la homocedasticidad entre niños y niñas. Vimos en el apartado anterior que esta variable no sigue una distribución normal para estos grupos, luego, para realizar nuestro análisis utilizaremos el test de Fligner-Killeen.

```
fligner.test(score ~ sex, data = students)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  score by sex
## Fligner-Killeen:med chi-squared = 0.18014, df = 1, p-value =
## 0.6713
```

Vemos, en este caso, que el valor de p obtenido (p-value) es mayor que 0,05. Luego, esto indica que no se observa diferencia significativa entre las varianzas por grupos de sexo.

Veamos ahora cómo se distribuye la varianza para la media escolar entre los alumnos clasificados en función de si reciben clases particulares o no. Nuevamente, como estos valores no siguen una distribución normal, utilizaremos el test de Fligner-Killeen

```
fligner.test(score ~ paid, data = students)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  score by paid
## Fligner-Killeen:med chi-squared = 0.95484, df = 1, p-value =
## 0.3285
```

Vemos que el valor de p es mayor de 0.05, luego, podemos concluir que las varianzas están homogéneamente distrubuidas.

Si ahora analizamos las distribuciones atendiendo a si reciben clases extraescolares o ayudas de sus padres, vamos a ver qué obtenemos. Igual que antes, aplicamos el test de Fligner-Killeen por la falta de normalidad en nuestros datos.

```
fligner.test(score ~ schoolsup, data = students)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  score by schoolsup
## Fligner-Killeen:med chi-squared = 13.884, df = 1, p-value =
## 0.0001945
```

En este caso, vemos que el valor de p obtenido es menor que 0,05 luego, vemos que la variable score (que recoge las medias de todas las notas obtenidas) presenta varianzas estadísticamente diferentes para los grupos de alumnos que reciben ayuda extra en el colegio.

Pasamos ahora a analizar cómo se distribuye la varianza de las notas medias para los grupos de alumnos que reciben ayuda suplementaria en casa. Usando, nuevamente, el test de Fligner-Killeen obtenemos:

```
fligner.test(score ~ famsup, data = students)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  score by famsup
## Fligner-Killeen:med chi-squared = 1.0762, df = 1, p-value = 0.2996
```

En este caso, se observa que el valor de p (p-value) es mayor de 0,05 luego podemos concluir que las varianzas son homogéneas para estos grupos de estudiantes

Analicemos cómo se distribuye la varianza de las notas medias por los diferentes grupos de edad, entre los alumnos menores de 16 y los mayores de 16

```
fligner.test(x = list(students.menores$score,students.mayores$score))
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  list(students.menores$score, students.mayores$score)
## Fligner-Killeen:med chi-squared = 5.3901, df = 1, p-value =
## 0.02025
```

El valor de p que obtenemos es menor que 0.05 luego, rechazamos la hipótesis de homocedasticidad y concluimos que la variable score tiene varianzas estadísticamente diferentes para los alumnos menores de 16 y los mayores de 16.

Por último, estudiaremos la distribución de la varianza para los grupos de alumnos que estudian más o menos de 3 horas.

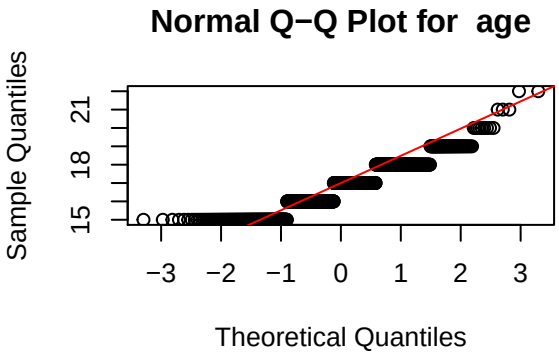
```
fligner.test(x = list(students.nostudytime$score,students.studytime$score))
```

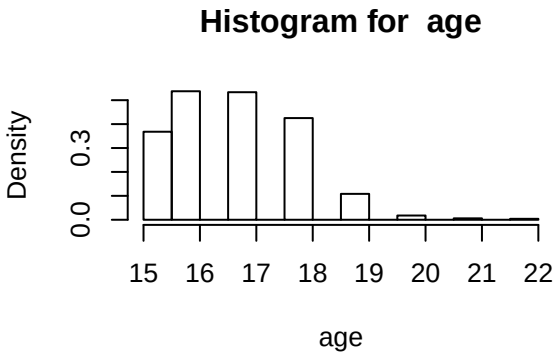
```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  list(students.nostudytime$score, students.studytime$score)
## Fligner-Killeen:med chi-squared = 0.0058552, df = 1, p-value =
## 0.939
```

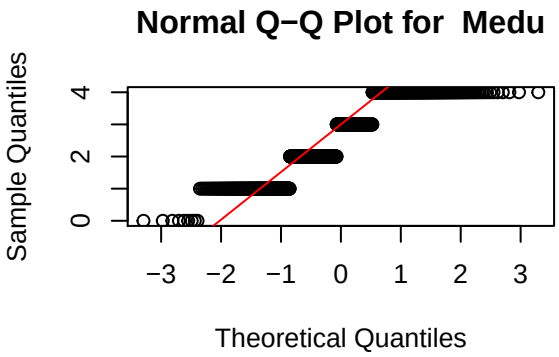
El valor de p es mayor de 0.05 luego podemos concluir que sí existe homogeneidad de la varianza

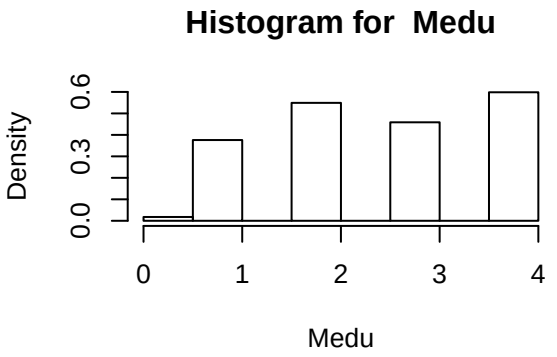
En las siguientes graficas mostramos como se traducen estos resultados visualmente. Para lo cual representamos las gráficas de quantile-quantile y los histogramas para cada una de nuestras variables.

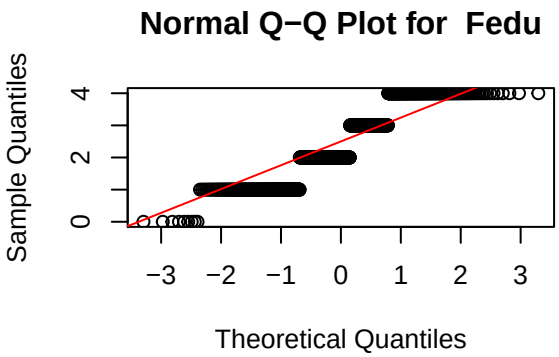
```
plotNormHistogram <- function(data, name) {
    qqnorm(data,main = paste("Normal Q-Q Plot for ",name))
    qqline(data,col="red")
    hist(data,
      main=paste("Histogram for ", name),
      xlab=name, freq = FALSE)
}

par(mfrow=c(2,2))
for(i in 1:ncol(students)) {
  if (is.numeric(students[,i])){
    plotNormHistogram(students[,i], colnames(students)[i])
  }
}
```

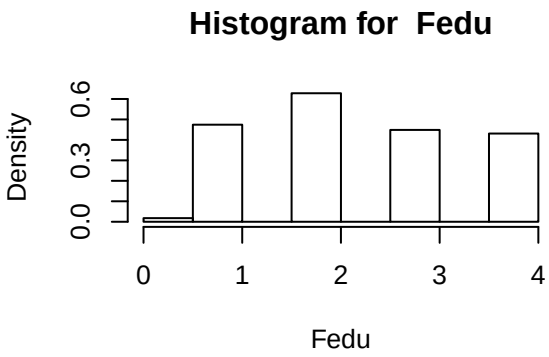

Normal Q−Q Plot for age
Sample Quantiles
15
18
21
−3
−2
−1
0
1
2
3
Theoretical Quantiles

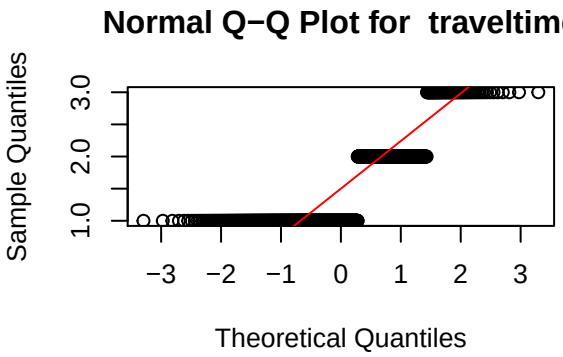

Histogram for age
Density
0.0
0.3
15
16
17
18
19
20
21
22
age

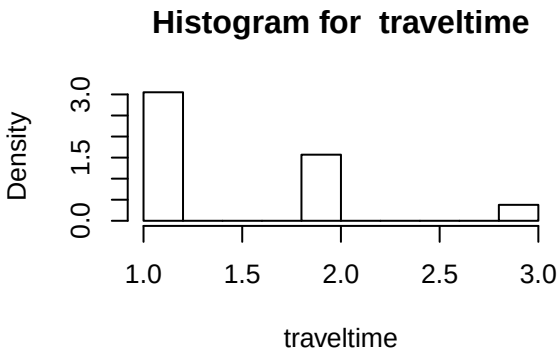

Normal Q−Q Plot for Medu
Sample Quantiles
0
2
4
−3
−2
−1
0
1
2
3
Theoretical Quantiles

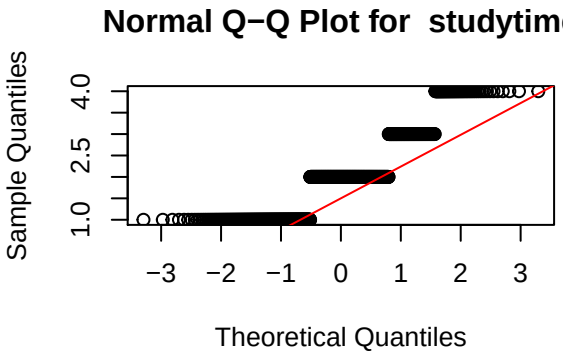

Histogram for Medu
Density
0.0
0.3
0.6
0
1
2
3
4
Medu

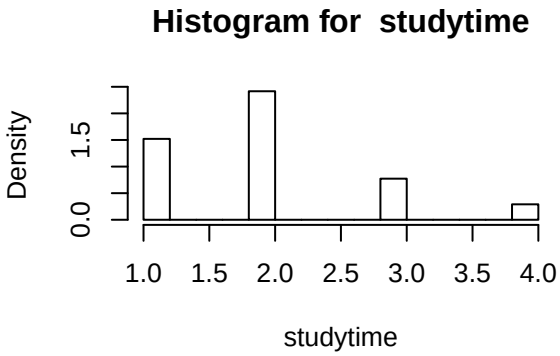

Normal Q−Q Plot for Fedu
Sample Quantiles
Theoretical Quantiles
0
2
4
−3
−2
−1
0
1
2
3

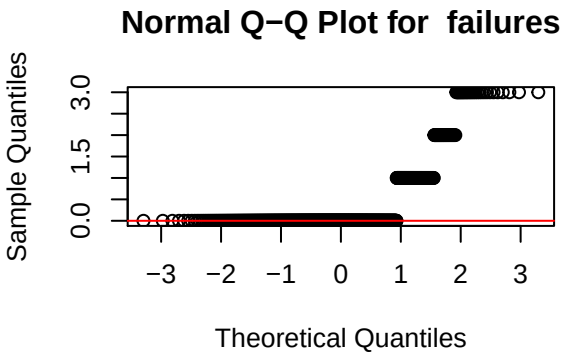

Histogram for Fedu
Density
Fedu
0.0
0.3
0.6
0
1
2
3
4

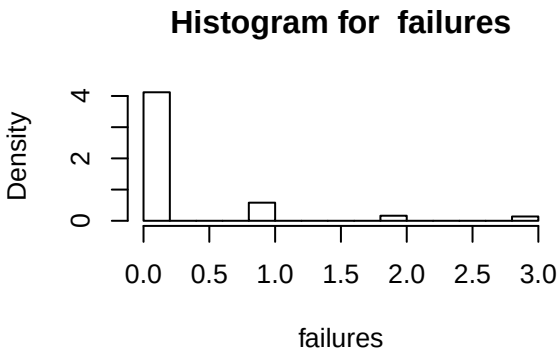

Normal Q−Q Plot for traveltime
Sample Quantiles
Theoretical Quantiles
1.0
2.0
3.0
−3
−2
−1
0
1
2
3

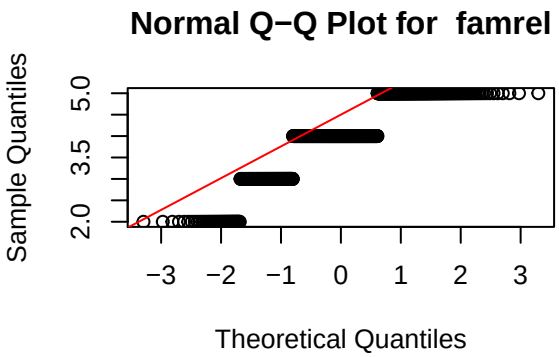

Histogram for traveltime
Density
traveltime
0.0
1.5
3.0
1.0
1.5
2.0
2.5
3.0

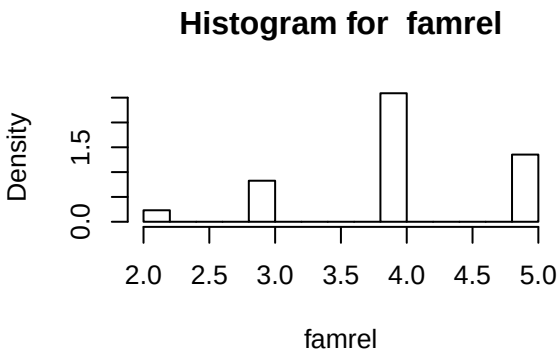

Normal Q−Q Plot for studytime
Sample Quantiles
Theoretical Quantiles
1.0
2.5
4.0
−3
−2
−1
0
1
2
3

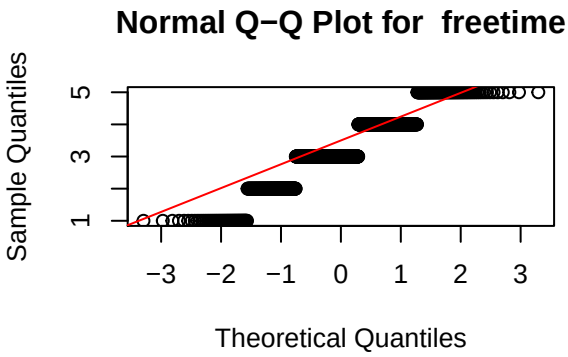

Histogram for studytime
Density
studytime
0.0
1.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0

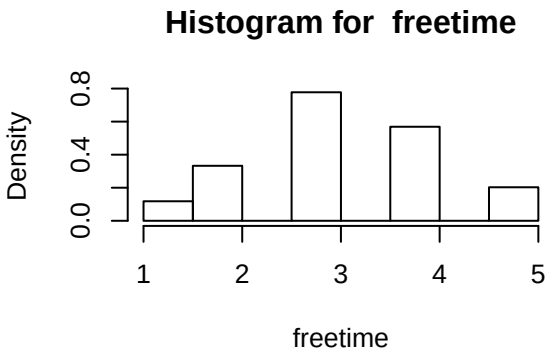

Normal Q−Q Plot for failures
Sample Quantiles
Theoretical Quantiles
0.0
1.5
3.0
−3
−2
−1
0
1
2
3

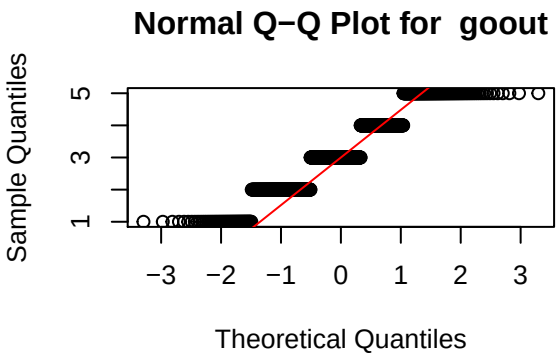

Histogram for failures
Density
failures
0
2
4
0.0
0.5
1.0
1.5
2.0
2.5
3.0

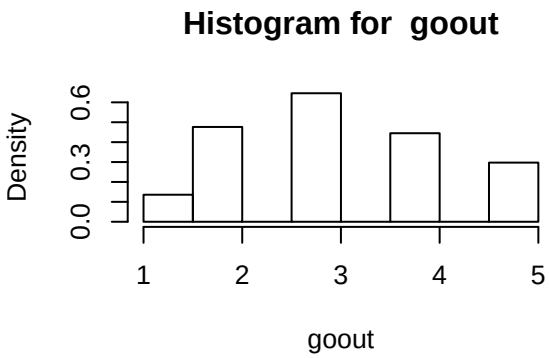

Normal Q−Q Plot for famrel
Sample Quantiles
2.0
3.5
5.0
−3
−2
−1
0
1
2
3
Theoretical Quantiles

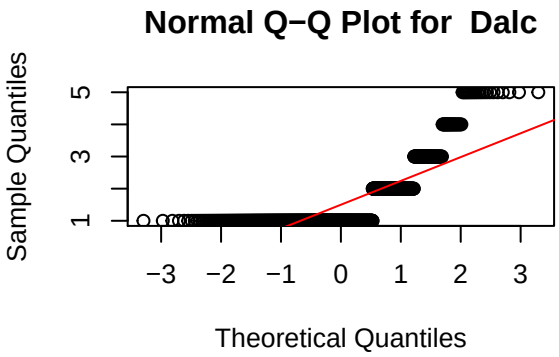

Histogram for famrel
Density
0.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
famrel

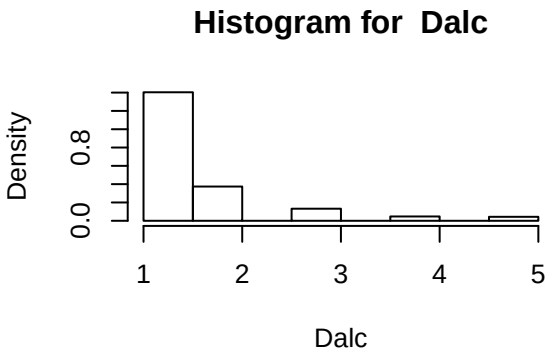

Normal Q−Q Plot for freetime
Sample Quantiles
1
3
5
−3
−2
−1
0
1
2
3
Theoretical Quantiles

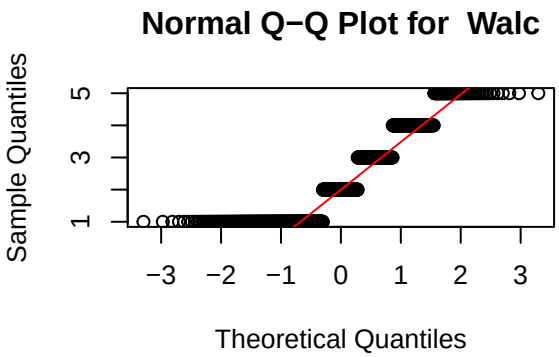

Histogram for freetime
Density
0.0
0.4
0.8
1
2
3
4
5
freetime

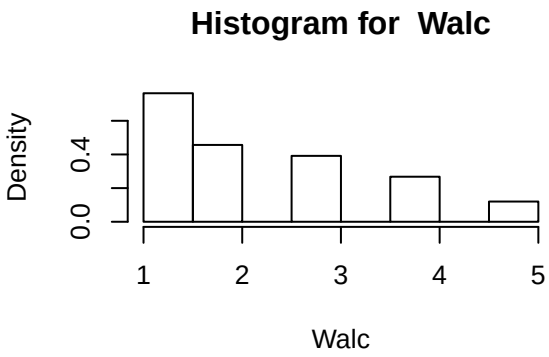

Normal Q−Q Plot for goout
Sample Quantiles
Theoretical Quantiles
1
3
5
−3
−2
−1
0
1
2
3

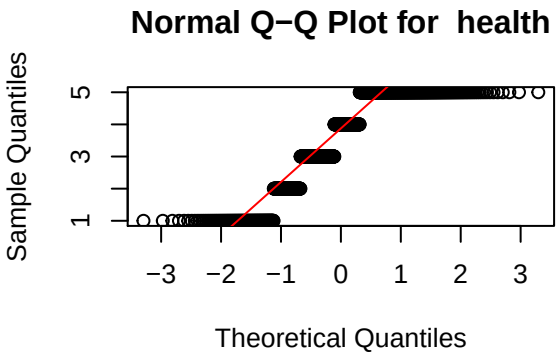

Histogram for goout
Density
goout
0.0
0.3
0.6
1
2
3
4
5

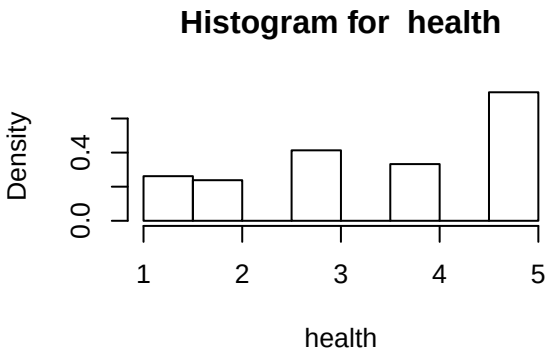

Normal Q−Q Plot for Dalc
Sample Quantiles
Theoretical Quantiles
1
3
5
−3
−2
−1
0
1
2
3

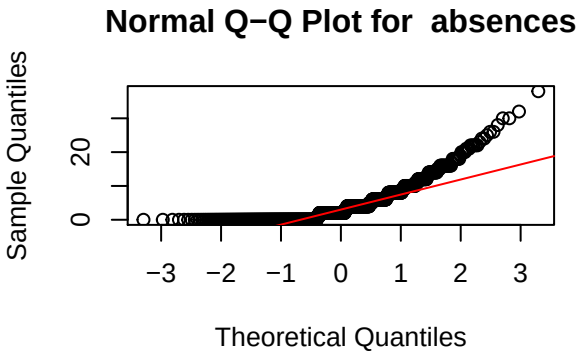

Histogram for Dalc
Density
Dalc
0.0
0.8
1
2
3
4
5

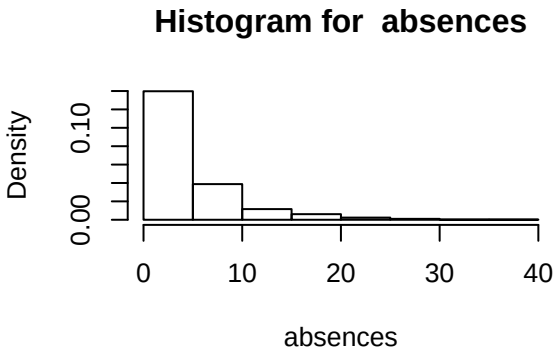

Normal Q−Q Plot for Walc
Sample Quantiles
Theoretical Quantiles
1
3
5
−3
−2
−1
0
1
2
3

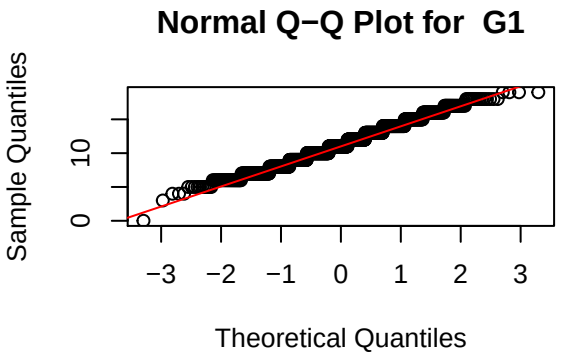

Histogram for Walc
Density
Walc
0.0
0.4
1
2
3
4
5

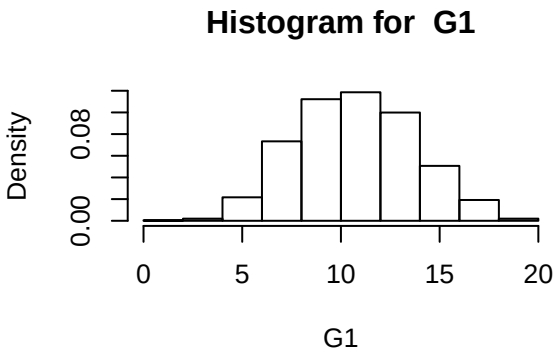

Normal Q−Q Plot for health
Sample Quantiles
Theoretical Quantiles
1
3
5
−3
−2
−1
0
1
2
3

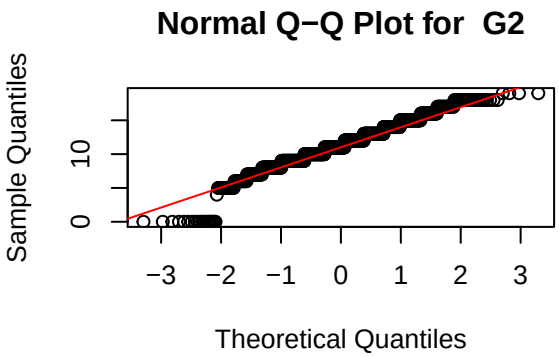

Histogram for health
Density
health
0.0
0.4
1
2
3
4
5

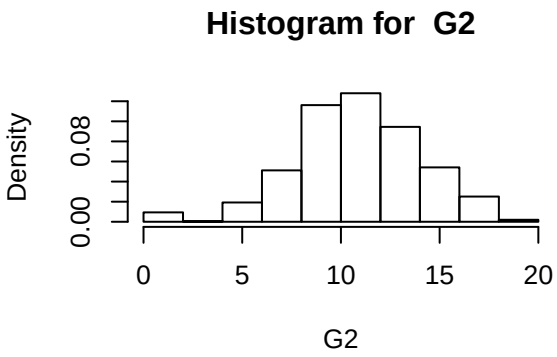

Normal Q−Q Plot for absences
Sample Quantiles
0
20
−3
−2
−1
0
1
2
3
Theoretical Quantiles

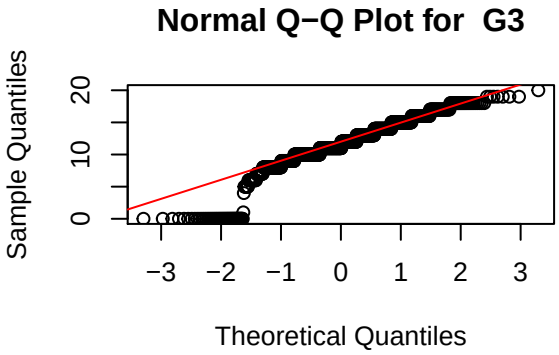

Histogram for absences
Density
0.00
0.10
0
10
20
30
40
absences

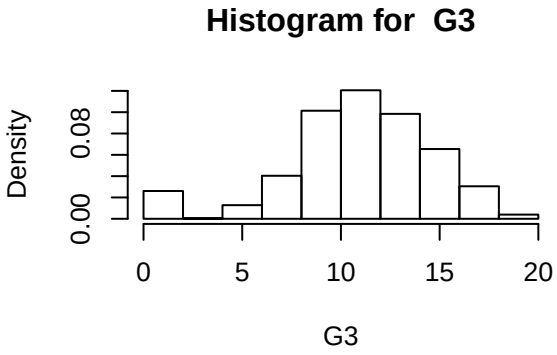

Normal Q−Q Plot for G1
Sample Quantiles
0
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles

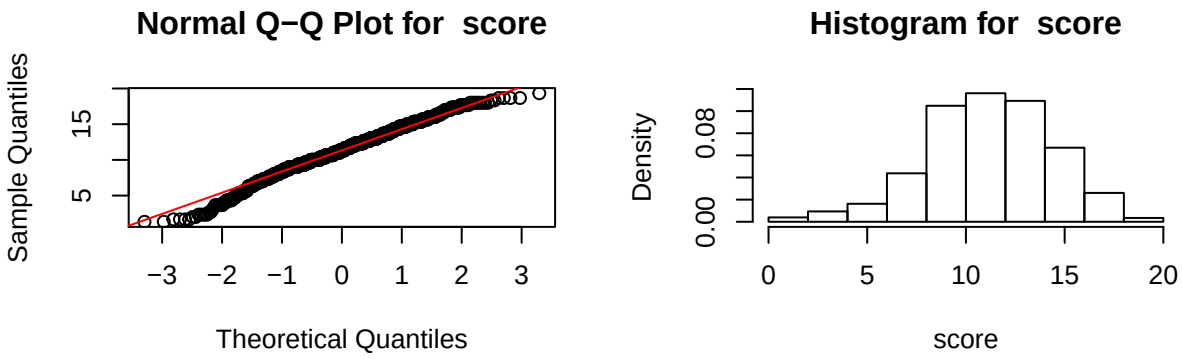

Histogram for G1
Density
0.00
0.08
0
5
10
15
20
G1


Normal Q−Q Plot for G2
Sample Quantiles
0
10
−3
−2
−1
0
1
2
3
Theoretical Quantiles


Histogram for G2
Density
0.00
0.08
0
5
10
15
20
G2

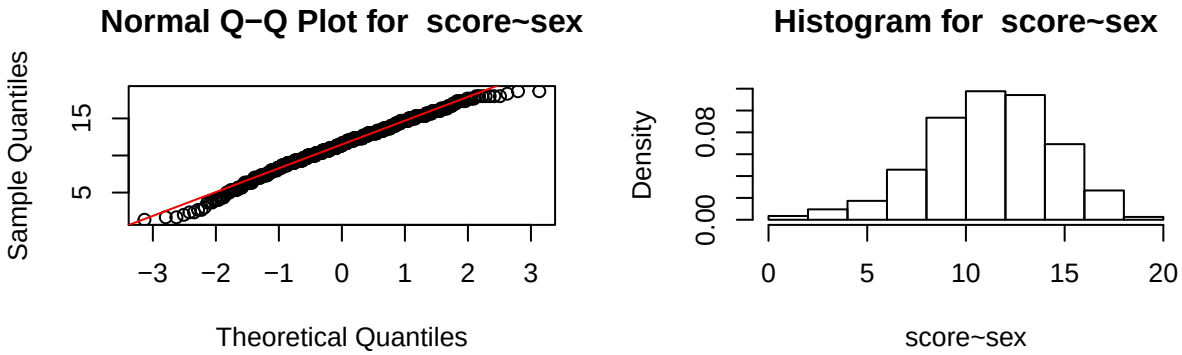

Normal Q−Q Plot for G3
Sample Quantiles
0
10
20
−3
−2
−1
0
1
2
3
Theoretical Quantiles

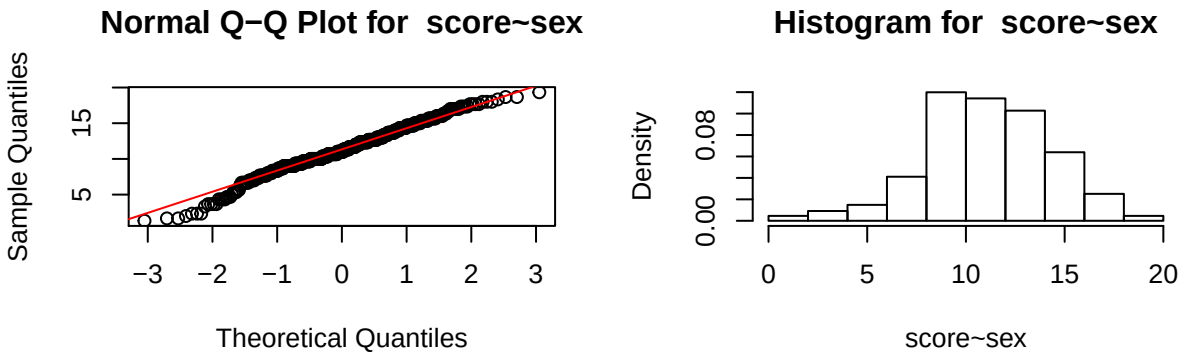

Histogram for G3
Density
0.00
0.08
0
5
10
15
20
G3



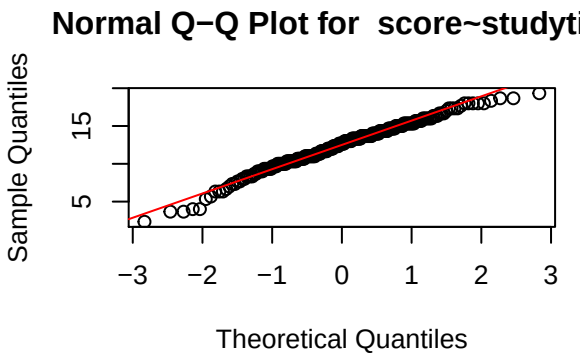
Y para la variables *score* de cada una de los grupos de estudio que hemos considerado.

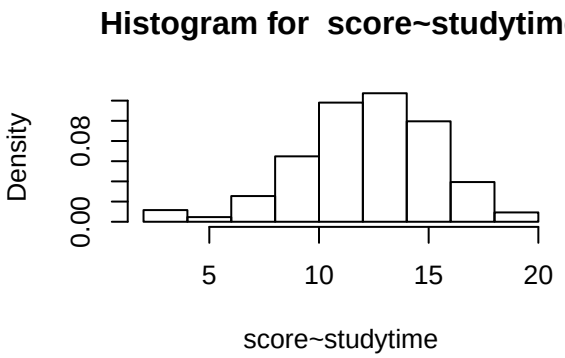
```
par(mfrow=c(2,2))
plotNormHistogram(students.female[,c("score")], "score~sex")
plotNormHistogram(students.male[,c("score")], "score~sex")
```

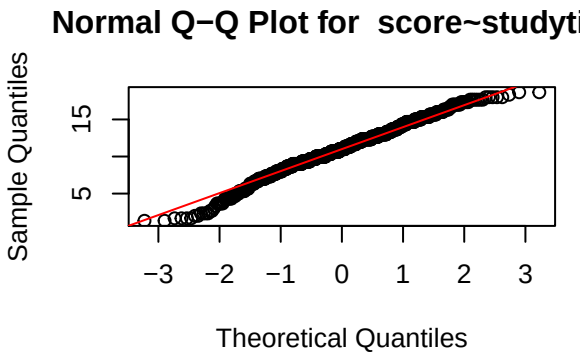


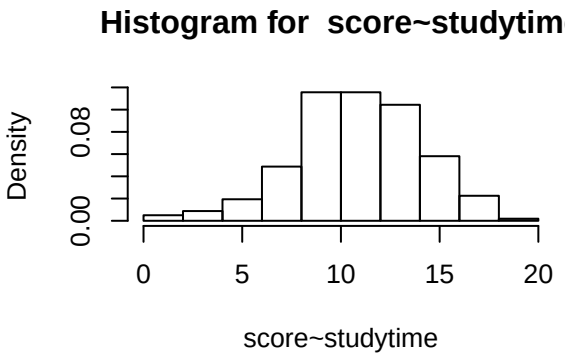



```
plotNormHistogram(students.studytime[,c("score")], "score~studytime")
plotNormHistogram(students.nostudytime[,c("score")], "score~studytime")
```

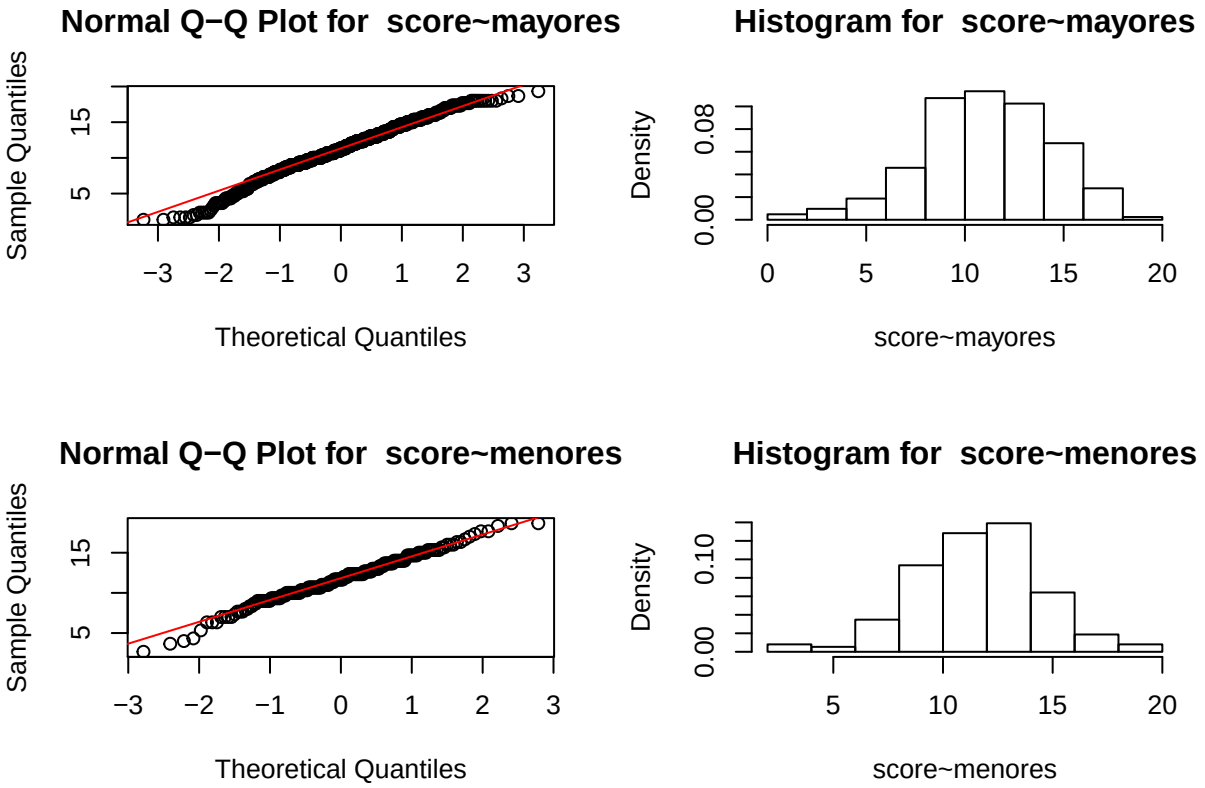









```
plotNormHistogram(students.mayores[,c("score")], "score~mayores")
plotNormHistogram(students.menores[,c("score")], "score~menores")
```



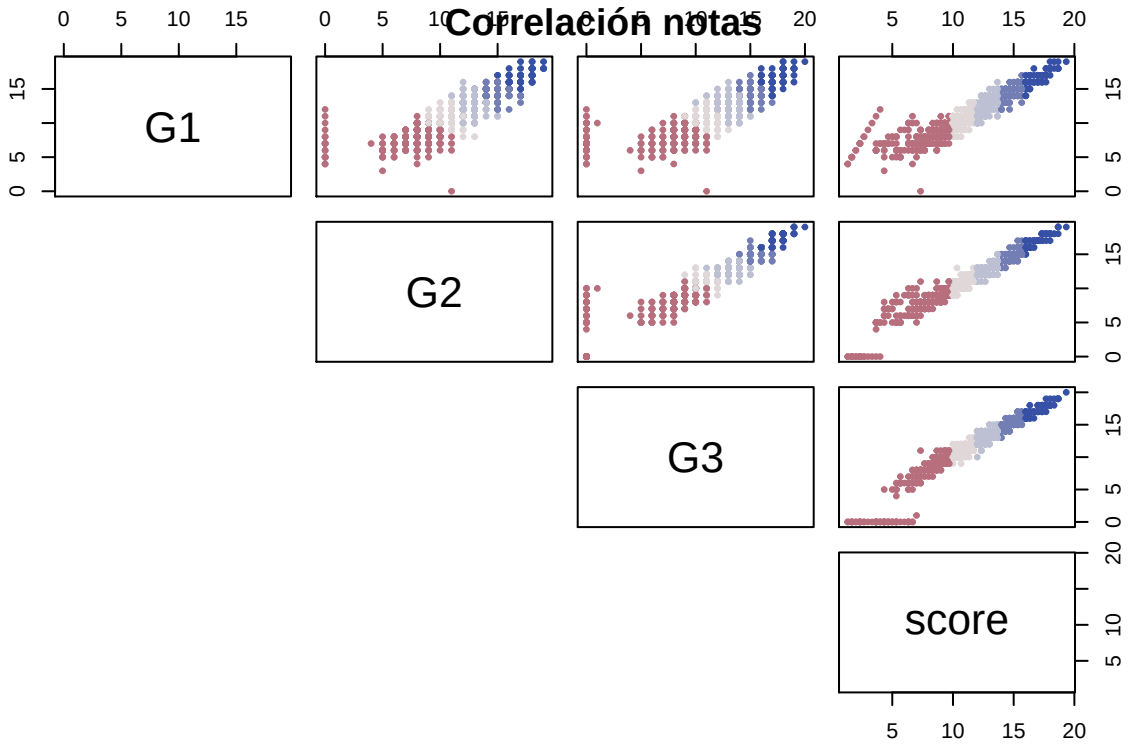
1.6 Pruebas estadísticas

En los siguientes apartados realizamos distintas pruebas estadísticas sobre nuestros conjuntos de datos. Hemos aplicado pruebas de contraste de hipótesis, correlaciones y regresiones para intentar responder a las preguntas que nos planteamos al inicio de este estudio.

1.6.1 Correlaciones

La siguiente gráfica muestra la correlación entre las distintas variables de notas:

```
color=diverge_hcl(length(students$calification))[rank(students$calification)]
pairs(~ G1 + G2 + G3 + score, data=students,pch=19,cex=0.5,lower.panel=NULL,col=color)
title("Correlación notas")
```



Luego, a la vista de estos diagramas, parece intuirse que existe una posible relación lineal entre las notas de cada trimestre y, con la media. Lo cual es evidente, ya que esta ha sido obtenida a partir de las otras tres.

Como ya hemos visto que estas variables no siguen una distribución normal, vamos a analizar la correlación utilizando el test de Spearman

```
cormat <- matrix(NA, nrow = 0, ncol = 2)
colnames(cormat) <- c("Function","spearman")
cormat <- rbind(cormat,c("G1 ~ G2", cor(x=students$G1,
                                        y=log10(students$G2), method="spearman")))
cormat <- rbind(cormat,c("G1 ~ G3", cor(x=students$G1,
                                        y=log10(students$G3), method="spearman")))
cormat <- rbind(cormat,c("G1 ~ score", cor(x=students$G1,
                                           y=log10(students$score), method="spearman")))
cormat <- rbind(cormat,c("G2 ~ G3", cor(x=students$G2,
                                        y=log10(students$G3), method="spearman")))
cormat <- rbind(cormat,c("G2 ~ score", cor(x=students$G2,
                                           y=log10(students$score), method="spearman")))
cormat <- rbind(cormat,c("G3 ~ score", cor(x=students$G3,
                                           y=log10(students$score), method="spearman")))
pander::pander(data.frame(cormat), split.table = 180)
```

Function	spearman
G1 ~ G2	0.899498870757684
G1 ~ G3	0.888119536070523
G1 ~ score	0.952311792863253
G2 ~ G3	0.955978495854601

Function	spearman
G2 ~ score	0.977437132075795
G3 ~ score	0.977616703850031

Vemos que para los 6 casos el grado de correlación es alto (mayor que 0,8). Pero para poder realmente considerar que existe correlación entre las variables debemos calcular la significancia.

```
cor.test(x = students$G1,
         y = log10(students$G2),
         alternative = "two.sided",
         conf.level  = 0.95,
         method      = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  students$G1 and log10(students$G2)
## S = 17567000, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.8994989
```

```
cor.test(x = students$G1,
         y = log10(students$G3),
         alternative = "two.sided",
         conf.level  = 0.95,
         method      = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  students$G1 and log10(students$G3)
## S = 19556000, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.8881195
```

```
cor.test(x = students$G1,
         y = log10(students$score),
         alternative = "two.sided",
         conf.level  = 0.95,
         method      = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  students$G1 and log10(students$score)
## S = 8335700, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9523118
```

```
cor.test(x = students$G2,
         y = log10(students$G3),
         alternative = "two.sided",
         conf.level  = 0.95,
         method      = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  students$G2 and log10(students$G3)
## S = 7694700, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9559785
```

```
cor.test(x = students$G2,
         y = log10(students$score),
         alternative = "two.sided",
         conf.level  = 0.95,
         method      = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  students$G2 and log10(students$score)
## S = 3943900, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9774371
```

```
cor.test(x = students$G3,
         y = log10(students$score),
         alternative = "two.sided",
         conf.level  = 0.95,
         method      = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  students$G3 and log10(students$score)
## S = 3912500, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9776167
```

Vemos que, para todos los casos, los coeficientes de correlación son significativos, puesto que p está próximo a 0 en los seis casos.

1.6.2 Pruebas de hipótesis

En esta sección, vamos a intentar responder a algunas de las preguntas que se plantearon en los objetivos iniciales mediante una prueba de hipótesis. Puesto que para las variables sobre las que vamos a realizar esta prueba hemos visto que no siguen los criterios de normalidad y homocedasticidad, exigidos para poder

aplicar la prueba de t-student, vamos a aplicar pruebas no paramétricas como Wilcoxon (cuando se comparen datos dependientes) o Mann-Whitney (cuando los grupos de datos sean independientes). Podemos aplicar esta prueba utilizando la función "*wilcox.text*".

1.6.2.1 ¿Las chicas sacan mejores notas que los chicos?

En la primera pregunta que nos planteabamos, de si las chicas sacan mejores notas que los chicos, la hipótesis consiste, en este caso, en considerar si las calificaciones obtenidas por las chicas y los chicos tienen idéntica distribución.

```
wilcox.test(score ~ sex, data = students)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  score by sex
## W = 131100, p-value = 0.3296
## alternative hypothesis: true location shift is not equal to 0
```

Por lo tanto, como vemos que p-value > 0.05 concluimos que las calificaciones obtenidas por los chicas son estadísticamente comparables a las obtenidas por los chicos. Consecuentemente, no podemos afirmar que las chicas sacan mejores notas que los chicos.

1.6.2.2 ¿Quien más tiempo dedica al estudio saca mejores notas?

En esta ocasión queremos comparar entre los estudiantes que dedican bastante tiempo semanalmente al estudio, y los que dedican menos tiempo.

Para esto, lo primero que vamos a hacer es crear una nueva variable categórica que nos permita distinguir entre los dos grupos mencionados arriba.

```
students$studytime2g<-students$studytime
students$studytime2g[students$studytime<3] <- "few"
students$studytime2g[students$studytime>=3] <- "many"
students$studytime2g <- as.factor(students$studytime2g)
```

Aplicando, ahora, la prueba sobre estos grupos de estudiantes obtenemos:

```
wilcox.test(score ~ studytime2g, data = students)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  score by studytime2g
## W = 63946, p-value = 4.336e-09
## alternative hypothesis: true location shift is not equal to 0
```

El valor de p-value obtenido es menor de 0.05. Por lo tanto, concluimos que ambos grupos de estudiantes presentan distribuiones muy distintas. Es decir, existe diferencias significativas en la notas obtenidas entre los estudiantes que dedican muchas horas al estudio y los que dedican pocas.

Si ahora comparamos entre los grupos de estudiantes que dedican 3 horas semanales y los que dedican 4 o más, vemos que el valor p-value es igual a 0.683. Es decir, es mayor que 0.05. Por lo tanto, siguen una misma distribución.

```
wilcox.test(score ~ studytime, data = students, subset = studytime %in% c(3, 4))
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
```

```
## data:  score by studytime
## W = 4800, p-value = 0.6813
## alternative hypothesis: true location shift is not equal to 0
```

1.6.2.3 ¿Aquellos alumnos que van a clases particulares o reciben ayuda por parte de sus padres sacan mejores notas?

Por último, vamos a aplicar la prueba de hipotésis para intentar responder a la otra pregunta que nos habíamos planteado: ¿hay diferencias en las calificaciones según los estudiantes reciban ayuda extra o no?

La primera prueba la aplicamos sobre los estudiantes que van a clases particulares y los que no:

```
wilcox.test(score ~ paid, data = students)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  score by paid
## W = 93904, p-value = 0.05975
## alternative hypothesis: true location shift is not equal to 0
```

En este caso, el valor de p es mayor, aunque por muy poco, a 0.05. Por lo tanto, diremos que no existe diferencia estadística en las notas, entre los alumnos que van a clases particulares y los que no.

En lo que se refiere a la distribución de las notas de los alumnos en función de si reciben ayuda extra por parte del colegio o no, ejecutando el test:

```
wilcox.test(score ~ schoolsup, data = students)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  score by schoolsup
## W = 63165, p-value = 3.59e-05
## alternative hypothesis: true location shift is not equal to 0
```

Entonces, puesto que $p < 0.05$, tenemos que existe diferencia significativa entre los alumnos que reciben ayuda extra y los que no (tal y como también vimos en el análisis visual de los datos).

Por último, aplicaremos esta prueba también sobre los grupos de estudiantes que son ayudados por su familia en casa y los que no.

```
wilcox.test(score ~ famsup, data = students)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  score by famsup
## W = 123890, p-value = 0.7474
## alternative hypothesis: true location shift is not equal to 0
```

Según el resultado obtenido, no podemos decir que exista diferencia significativa en las notas finales, entre los estudiantes que son ayudados por su familia en casa y los que no.

1.6.3 Modelo de regresión lineal

Tal y como hemos visto, existe una fuerte correlación entre las distintas variables que se refieren a las calificaciones y la nota final. Por lo tanto, vamos a utilizar estas variables para obtener un modelo lineal que nos permita predecir las notas de los estudiantes a partir de algunas de las otras variables.

En primer lugar, dividiremos nuestros datos en un conjunto de datos de entrenamiento y otro de datos de prueba. El primero nos servirá para generar los distintos modelos y, el segundo, nos permitirá evaluar la precisión de las predicciones realizadas por nuestro modelo.

```
sample_size = floor(0.90*nrow(students)) # 90% train - 10% test
train_idx = sample(seq_len(nrow(students)),size = sample_size)
train = students[train_idx,]
test = students[-train_idx,]
```

Definiremos varios modelos utilizando distintas variables relacionadas con la calificación final (*score*) para, finalmente elegir uno que utilizaremos para predecir la nota final para los estudiantes en nuestro conjunto de test.

```
# Generación de varios modelos
modelo1 <- lm(score ~ G1 + G2 + G3, data = train)
modelo2 <- lm(score ~ G1 + G2, data = train)
modelo3 <- lm(score ~ G1 + G3, data = train)
modelo4 <- lm(score ~ G1, data = train)
modelo5 <- lm(score ~ G2, data = train)
modelo6 <- lm(score ~ G3, data = train)
modelo7 <- lm(score ~ G1 + G3 + studytime, data = train)
modelo8 <- lm(score ~ G2 + studytime + schoolsup, data = train)
modelo9 <- lm(score ~ studytime + sex + absences, data = train)
modelo10 <- lm(score ~ studytime + paid, data = train)

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,
      9, summary(modelo9)$r.squared,
      10, summary(modelo10)$r.squared),
    ncol = 2, byrow = TRUE)
colnames(tabla.coeficientes) <- c("Modelo", "R^2")
tabla.coeficientes
```

```
##       Modelo        R^2
##  [1,]      1 1.00000000
##  [2,]      2 0.97378181
##  [3,]      3 0.98559872
##  [4,]      4 0.85890014
##  [5,]      5 0.94148757
##  [6,]      6 0.92339449
##  [7,]      7 0.98560812
##  [8,]      8 0.94234186
##  [9,]      9 0.03377100
## [10,]     10 0.03471087
```

Esta claro, según confirmamos por los resultados obtenidos en esta tabla, que si utilizamos las notas de los tres trimestres para predecir la calificación final el resultado será exacto, ya que esta nota se obtiene a partir de un calculo directo sobre las otras tres (la media). Ignorando este caso, analizamos el resultado obtenido para los otros modelos, en los cuales se quiere predecir la nota final utilizando alguna otra variable o conjunto

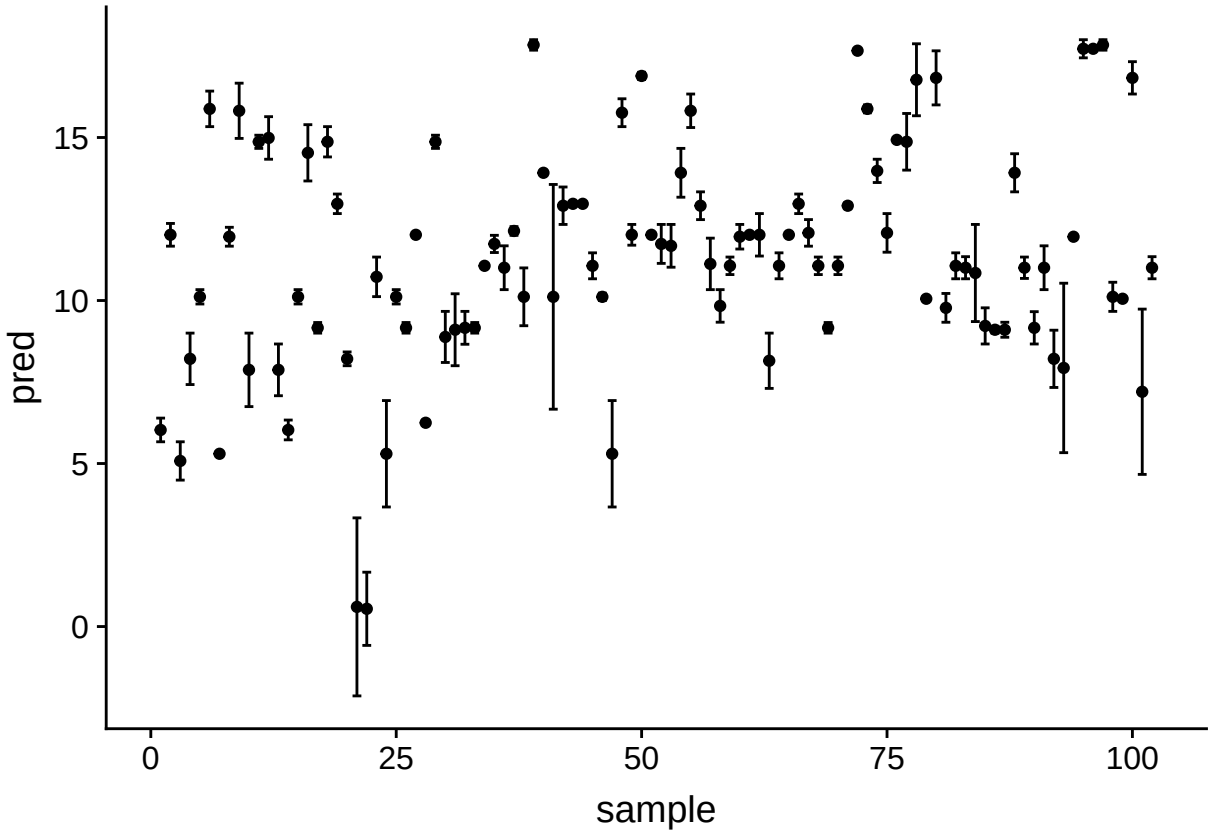
de variables.

De la tabla anterior, elegimos el modelo 8, que nos permitiría predecir la nota final a partir de la nota del segundo trimestre y las variables categóricas *studytime* y *schoolsup*. Utilizando este modelo y el conjunto de datos de test que habiamos reservado podemos predecir las notas finales para estos estudiantes.

```
y_predict = predict(modelo8, test)
```

En la siguiente gráfica mostramos el error cometido en cada una de las predicciones:

```
data = data.frame(G2=test$score, pred=y_predict, se=abs(y_predict - test$score))
data$sample <- seq.int(nrow(data))
ggplot(data, aes(x=sample, y=pred)) +
    geom_errorbar(aes(ymin=pred-se, ymax=pred+se)) +
    geom_point()
```



1.6.4 Regresión logística

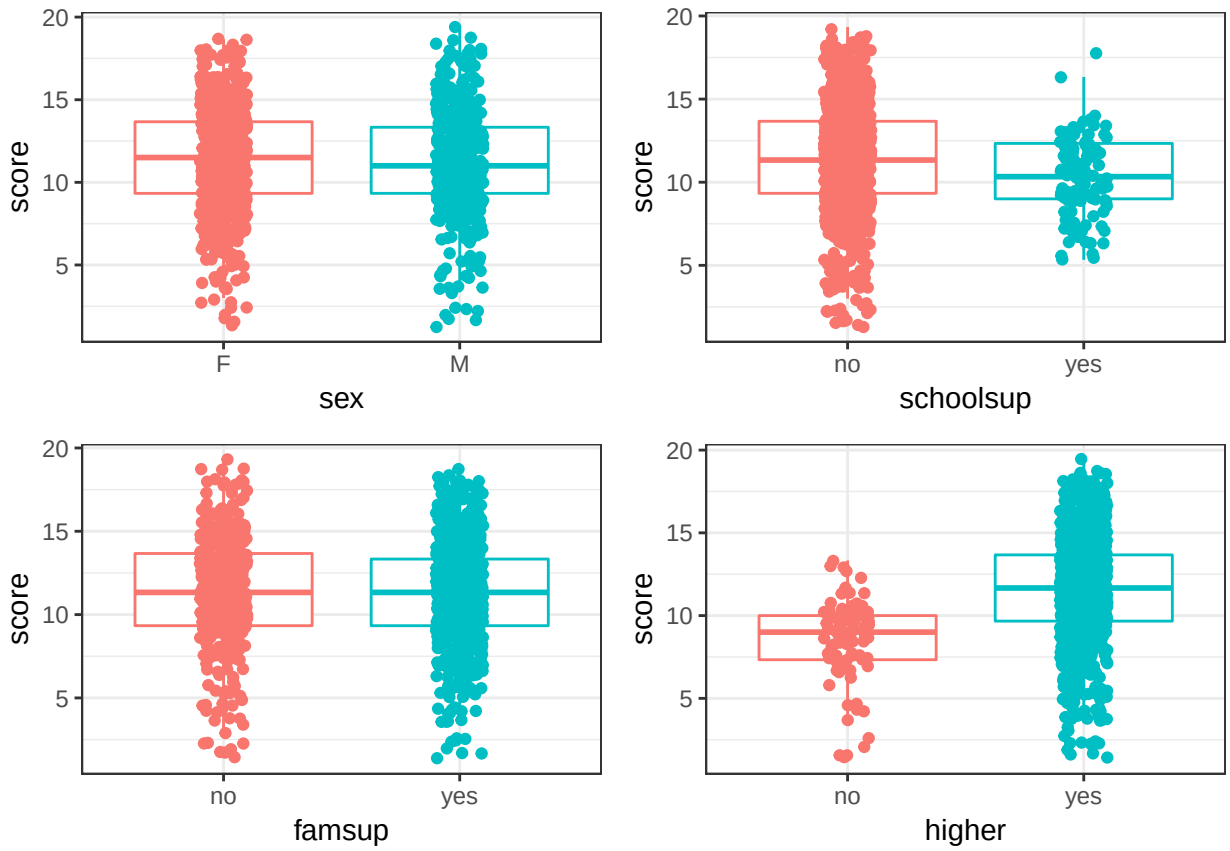
Vamos a utilizar regresión logística para intentar averiguar qué variable, de las categóricas con las que hemos estado trabajando, puede predecirse mejor a partir de la nota media del curso. Vamos a generar modelos de regresión logística para distintas variables dicotómicas, y veremos cuál de ellos es mejor.

Nos centraremos en las siguientes variables:

- Sexo: ¿podemos predecir el sexo del alumno en función de la nota?
- SchoolSup: ¿podemos predecir si el alumno ha tenido ayuda en la escuela en función de la nota media?
- famSup: ¿podemos predecir si el alumno ha tenido ayuda en casa en función de la nota media?
- higher: ¿podemos predecir si el alumno querrá seguir estudiando en función de la nota media obtenida?

En primer lugar, representamos las observaciones para cada uno de estos casos para poder intuir gráficamente si la variable escogida (la media de las notas: score) está relacionada con la variable respuesta (cualquiera de las anteriores) y, por lo tanto, puede considerarse que es un buen predictor.

```
fig1 = ggplot(data = students, aes(x = sex, y = score, color = sex)) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(width = 0.1) +
  theme_bw() +
  theme(legend.position = "null")

fig2 = ggplot(data = students, aes(x = schoolsup, y = score, color = schoolsup)) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(width = 0.1) +
  theme_bw() +
  theme(legend.position = "null")

fig3 = ggplot(data = students, aes(x = famsup, y = score, color = famsup)) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(width = 0.1) +
  theme_bw() +
  theme(legend.position = "null")

fig4 = ggplot(data = students, aes(x = higher, y = score, color = higher)) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(width = 0.1) +
  theme_bw() +
  theme(legend.position = "null")

grid.arrange(fig1, fig2, fig3, fig4,ncol=2)
```



A la vista de las gráficas, parece que en el único caso en el que se aprecian diferencias significativas en las notas es en el caso en el que los estudiantes quieren seguir estudiando (variable *higher*).

Ahora vamos a generar los modelos para cada una de ellas y ver qué obtenemos

```
m1 <- glm(sex ~ score, data = students, family = "binomial")
summary(m1)
```

```
##
## Call:
## glm(formula = sex ~ score, family = "binomial", data = students)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.119  -1.064  -1.041   1.293   1.346
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.12144    0.22984  -0.528    0.597
## score       -0.01380    0.01957  -0.705    0.481
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1389.1  on 1015  degrees of freedom
## Residual deviance: 1388.6  on 1014  degrees of freedom
## AIC: 1392.6
##
```

```
## Number of Fisher Scoring iterations: 4
```

```
m2 <- glm(schoolsup ~ score, data = students, family = "binomial")
summary(m2)
```

```
##
## Call:
## glm(formula = schoolsup ~ score, family = "binomial", data = students)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7553  -0.5148  -0.4604  -0.4047   2.3777
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.97320    0.33213  -2.930 0.003388 **
## score       -0.10147    0.03021  -3.359 0.000783 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 709.29  on 1015  degrees of freedom
## Residual deviance: 698.08  on 1014  degrees of freedom
## AIC: 702.08
##
## Number of Fisher Scoring iterations: 5
```

```
m3 <- glm(famsup ~ score, data = students, family = "binomial")
summary(m3)
```

```
##
## Call:
## glm(formula = famsup ~ score, family = "binomial", data = students)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3861  -1.3772   0.9872   0.9895   0.9941
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.480656   0.234243   2.052   0.0402 *
## score       -0.001761   0.019911  -0.088   0.9295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1356.0  on 1015  degrees of freedom
## Residual deviance: 1355.9  on 1014  degrees of freedom
## AIC: 1359.9
##
## Number of Fisher Scoring iterations: 4
```

```
m4 <- glm(higher ~ score, data = students, family = "binomial")
summary(m4)
```

```
##
## Call:
## glm(formula = higher ~ score, family = "binomial", data = students)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5849   0.2226   0.3234   0.4261   1.2286
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.50010    0.34669  -1.443    0.149
## score        0.28537    0.03613   7.898 2.84e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 589.22  on 1015  degrees of freedom
## Residual deviance: 521.16  on 1014  degrees of freedom
## AIC: 525.16
##
## Number of Fisher Scoring iterations: 6
```

Se observa que el AIC (criterio de información de Akaike) menor de los 4 modelos es el correspondiente al hecho de querer cursar estudios superiores, tal y como se intuía en las gráficas.

1.7 Conclusiones

Hemos partido de un conjunto de datos de alumnos estudiantes de matemáticas y portugués en los que se recogía información de diferentes aspectos, como por ejemplo: el sexo de los alumnos, el número de suspensos, si reciben ayuda extra en sus estudios ya sea esta en el colegio o pagada por sus padres, si quieren continuar con sus estudios, y las notas obtenidas durante los tres trimestres, entre otros.

Partimos de dos ficheros csv, uno para los datos de estudiantes de matemáticas y otro para los de portugés. Como primer paso, hemos unido ambos conjuntos en un único data set añadiendo una nueva columna indicando la asignatura de la que provenían. Una vez que los teníamos juntos hemos creado una varibale numérica nueva con la media de todas las notas y otra variable categórica asociando esa nota media a una nota categorizada (A, B, C, D y F). Tras comprobar que no existian valores nulos ni vacíos, hemos analizado los outliers que había en cada variable numérica y hemos tratado particularmente cada uno de ellos, teniendo en cuenta la relevancia de los mismos para cada uno de los casos.

En primer lugar hemos analizado la correlación entre las notas: vemos que lo están. Es decir, que las notas de cada alumno en cada trimestre están determinadas fuertemente por las obtenidas en los anteriores o, dicho de otra forma, que un estudiante que saca buenas notas en el primer trimestre es muy probable que saque buenas notas en los siguientes.

Hemos analizado también si existe diferencias significativas, utilizando la prueba de hipótesis, entre las notas de los estudiantes en función de su sexo, las horas que dedican semanalmente al estudio o, si reciben algún tipo de ayuda extra en el colegio, en casa o a través de clases particulares. Hemos concluido que es importante dedicar al menos 3 horas semanales al estudio para obtener buenos resultados académicos. También hemos visto que existen diferencias significativas entre los alumnos que reciben ayuda extra por parte del colegio y los que no. En cambio, no existe diferenciación entre las notas de chicos y chicas, ni entre los estudiantes que van a clases particulares, ni los que reciben ayuda por parte de su familia.

Otro objetivo principal de nuestro estudio era saber si existe alguna variable en el juego de datos que pueda ayudarnos a predecir las notas. Para esto hemos aplicado un modelo de regresión lineal aprovechando la correlación existente entre las variables de notas.

Además, hemos usado regresión logística para poder saber si las notas pueden determinar alguna de las otras características. Hemos intentando averiguar cuál de las variables binarias que tenemos puede estar determinada por las notas obtenidas y vemos que, de todas las que hemos analizado, la más determinada es la voluntad del estudiante de continuar con cursos superiores. Es decir, que podemos ver que un estudiante que ha obtenido una nota alta es muy probable que continúe sus estudios. Luego parece lógico pensar que un alumno que quiere continuar estudiando tendrá buenas calificaciones.

2 Recursos

Los siguientes recursos son de utilidad para la realización de la práctica:

Calvo M., Subirats L., Pérez D. (2019). Introducción a la limpieza y análisis de los datos. Editorial UOC.

Megan Squire (2015). Clean Data. Packt Publishing Ltd.

Jiawei Han, Micheine Kamber, Jian Pei (2012). Data mining: concepts and techniques. Morgan Kaufmann.

Jason W. Osborne (2010). Data Cleaning Basics: Best Practices in Dealing with Extreme Scores. Newborn and Infant Nursing Reviews; 10 (1): pp. 1527-3369.

Peter Dalgaard (2008). Introductory statistics with R. Springer Science & Business Media.

Wes McKinney (2012). Python for Data Analysis. O'Reilley Media, Inc.

Tutorial de Github https://guides.github.com/activities/hello-world.