Práctica 3

Fundamentos de la Ciencia de Datos

Samuel Aós Paumard,
Enrique Coronado Barco,
Carmen Martínez Estévez,
Alberto Martínez Ortega

December 1, 2020

1 Ejericio 1

La primera parte consiste en la realización de dos ejercicios en clase con ayuda del profesor:

1.1 Apartado 1.1

En el primero se va a realizar un análisis de clasificación de datos con R aplicando los conceptos vistos en el tema. De la misma muestra que se utilizó para hacer de forma manual el primer ejercicio de clasificación para datos cualitativos, se deberá obtener, utilizando la medida de Ganancia de información, mediante la medida de impureza de Gini, la función de clasificación.

Lo primero es cargar las librerías a utilizar. En este caso: ggplot2 (para tratamiento de gráficos complejos), rpart (conjunto de árboles de decisión), rpart.plot(para poder hacer un plot correcto de los datos de rpart).

```
> install.packages('ggplot2', repos="https://cran.rediris.es/")
> install.packages('rpart', repos="https://cran.rediris.es/")
> install.packages('rpart.plot', repos="https://cran.rediris.es/")
> library('ggplot2')
> library('rpart')
> library("rpart.plot")
```

Cargamos los datos desde el archivo de texto 'calificaciones' y observamos dichos datos: tenemos las notas de Teoría, Laboratorio y Práctica y queremos decidir qué calificación global se espera en función de las notas anteriores.

```
> calificaciones <- read.table("calificaciones.txt")
> print(calificaciones)
```

```
        Teoria Lab Pract Calificacion
suceso1      A   A     B           Ap
suceso2      A   B     D           Ss
suceso3      D   D     C           Ss
suceso4      D   D     A           Ss
```

```
suceso5      B   C     B           Ss
suceso6      C   B     B           Ap
suceso7      B   B     A           Ap
suceso8      C   D     C           Ss
suceso9      B   A     C           Ss
```

Convertimos la tabla de calificaciones a dataframe para que después estén correctamente formateados para su tratamiento.

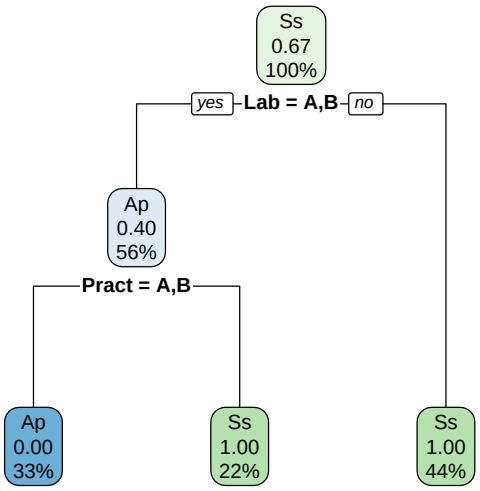
```
> muestra <- data.frame(calificaciones)
> print(muestra)

        Teoria Lab Pract Calificacion
suceso1      A   A     B           Ap
suceso2      A   B     D           Ss
suceso3      D   D     C           Ss
suceso4      D   D     A           Ss
suceso5      B   C     B           Ss
suceso6      C   B     B           Ap
suceso7      B   B     A           Ap
suceso8      C   D     C           Ss
suceso9      B   A     C           Ss
```

Utilizamos la función rpart para llevar a cabo la clasificación con los árboles de decisión. En concreto con el algoritmo de Hunt, vamos a construir uno. Le pasamos como parámetros el nombre de la función clasificadora, el conjunto de datos muestra, método class para la clasificación y el número mínimo de veces que hay que llegar a un nodo para que sea considerado.

```
> clasificacion <- rpart(Calificacion~., data=muestra, method="class", minsplit=1)
```

Esto nos permite visualizar la medida de impureza de Gini, usada por defecto en rpart y dibujar la gráfica con dichos datos:

```
> print(clasificacion)

n= 9

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 9 3 Ss (0.3333333 0.6666667)
  2) Lab=A,B 5 2 Ap (0.6000000 0.4000000)
    4) Pract=A,B 3 0 Ap (1.0000000 0.0000000) *
    5) Pract=C,D 2 0 Ss (0.0000000 1.0000000) *
  3) Lab=C,D 4 0 Ss (0.0000000 1.0000000) *

> rpart.plot(clasificacion)
>
```



Gini nos dice que si extraemos dos elementos de una población al azar, entonces deben ser de la misma clase y la probabilidad de esto es 1 si la población es pura. Dicho esto, y fijándonos en los resultados, el texto nos ofrece la misma información que la imágen, solo que esta de una forma más agradable y comprensible. Fijémonos en esta para continuar la explicación. Tal y como se puede ver, antes de comenzar la clasificación, la probabilidad de población pura era de 0.67. Si nos fijamos en las notas de laboratorio, todos aquellos que no tienen una A o B, están suspensos. Después seguimos con otro de los atributos y nos encontramos con que si no sacas una A o B en práctica, estás suspenso siempre. En cambio si las sacas, estás 100

1.2 Apartado 1.2

En el segundo, para los datos de un fichero .txt generado a partir de los datos de la muestra 2 utilizada para hacer el segundo ejercicio de clasificación para datos cuantitativos, hacer un análisis de regresión lineal. En el apartado anterior hacíamos un análisis sobre datos cualitativos (con unas determinadas propiedades que les clasifican). En esta parte vamos a ver datos cuantitativos correspondientes a los radios ecuatoriales y densidades de los planetas interiores. Cargamos los datos a utilizar:

```
> planetas<-read.table("planetas.txt")
> print(planetas)

           R   D
Mercurio 2.4 5.4
Venus    6.1 5.2
```

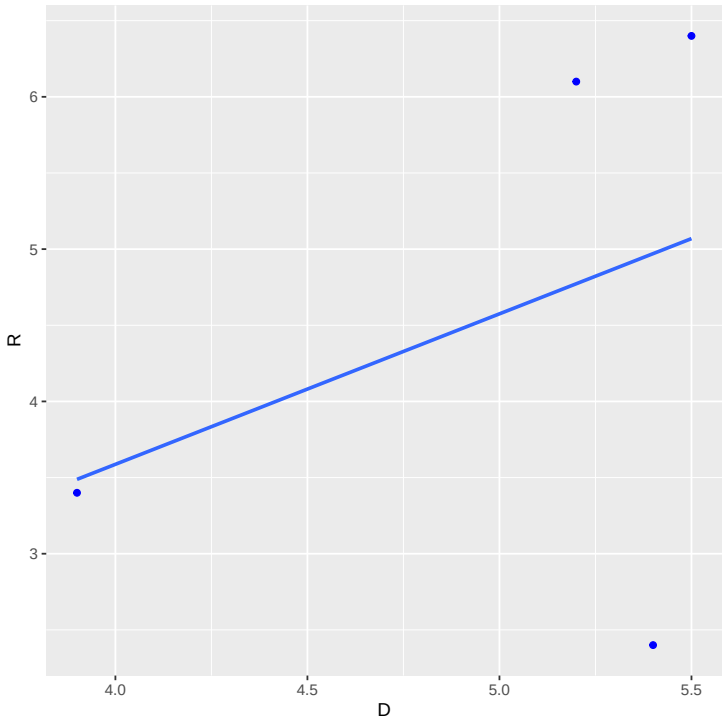
```
Tierra   6.4 5.5
Marte    3.4 3.9
```

Convertimos los datos a dataframe:

```
> muestra2<-data.frame(planetas)
> print(planetas)

           R   D
Mercurio 2.4 5.4
Venus    6.1 5.2
Tierra   6.4 5.5
Marte    3.4 3.9
```

Procedemos a hacer el análisis de regresión con los datos ya debidamente formateados. Lo hacemos con la función lm que se encarga de hacer análisis lineales en función de los parámetros pasados. Estos son: las variables dependiente e independiente y el conjunto de datos. Con la función ggplot exponemos estos resultados. Podemos observar en la gráfica la recta de regresión y los 4 puntos que representan los 4 planetas tomados para este ejemplo. La mala distribución de los puntos y su dispersión hace muy difícil que esta recta pase cerca de todos los puntos, es decir, se aproxime a ellos, y por tanto sea una buena función clasificadora.

```
> regresion<-lm(D~R, data=muestra2)
> ggplot(planetas, aes(x = D, y = R)) + geom_point(color='blue')+ geom_smooth(method = "lm
```



2 Ejericio 2

La segunda parte consiste en la realización de dos ejercicios:

2.1 Apartado 2.1

Desarrollo por parte de cada alumno del enunciado y la solución de un ejercicio en el que se realice un análisis con R de clasificación supervisada con árboles introduciendo modificaciones sobre el ejercicio hecho en clase.

Se he generado un archivo de texto con datos de alumnos referentes al número de asignaturas que estos cursan, su edad y el curso en el que se encuentran. El objetivo de las modificaciones introducidas es probar otros métodos de análisis y clasificación supervisada mediante árboles. Los métodos utilizados han sido:

- class
- poisson
- anova
- C50

Los paquetes *rpart* y *c50* serán necesarios para la ejecución de los distintos métodos, siendo rpart utilizado por los tres primeros y *C50* específico de su método. Además se carga la librería *rpart.plot* para una visualización más cómoda de los datos generados por rpart.

```
> install.packages("rpart",repos = "https://cran.rediris.es/")
> install.packages("rpart.plot",repos = "https://cran.rediris.es/")
> install.packages("C50",repos = "https://cran.rediris.es/")
> library(rpart)
> library(rpart.plot)
> library(C50)
```

Tras cargar los paquetes necesarios introduciremos en memoria los datos del fichero generado con la información del alumnado.

```
> alumnos = read.table("datos.txt")
> alumnos
```

```
   asignaturas edad curso
1           10   20     4
2            8   23     3
3           10   19     4
4            8   20     3
5            8   19     3
6            9   21     1
7           10   19     1
8            9   19     1
9            9   22     2
10           9   19     3
11          10   22     1
12           9   19     2
```

```
13           9   20     2
14          10   21     1
15           8   23     1
16           9   20     4
17           9   23     4
18           9   22     3
19          10   23     3
20          10   23     1
21           8   19     2
22           9   19     2
23           9   20     2
24          10   21     1
25           8   23     1
26           9   20     4
27           9   23     4
28           9   22     3
29          10   23     3
30          10   23     1
31           8   19     2
32           9   19     2
```

Para garantizar el correcto funcionamiento establecemos el conjunto de datos leídos como un cuadro de datos en nuestras variables.

```
> muestra = data.frame(alumnos)
> muestra
```

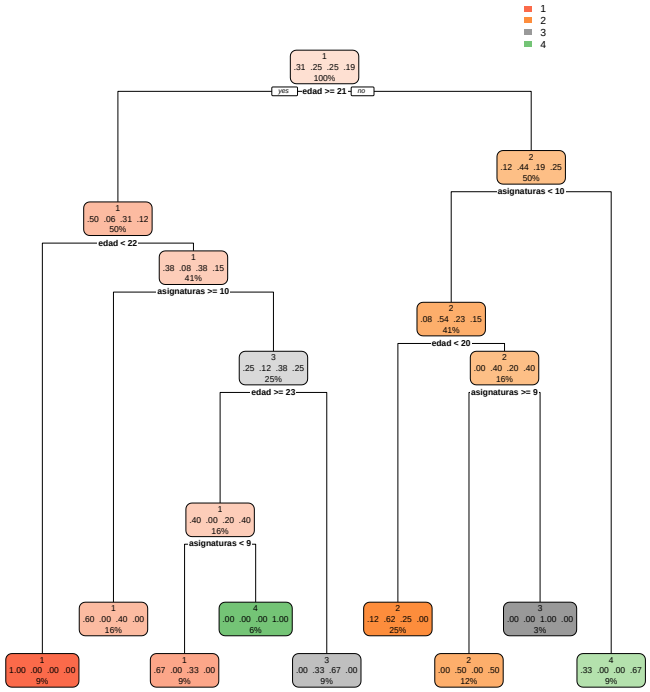
```
   asignaturas edad curso
1           10   20     4
2            8   23     3
3           10   19     4
4            8   20     3
5            8   19     3
6            9   21     1
7           10   19     1
8            9   19     1
9            9   22     2
10           9   19     3
11          10   22     1
12           9   19     2
13           9   20     2
14          10   21     1
15           8   23     1
16           9   20     4
17           9   23     4
18           9   22     3
19          10   23     3
20          10   23     1
21           8   19     2
22           9   19     2
23           9   20     2
```

```
24          10   21     1
25           8   23     1
26           9   20     4
27           9   23     4
28           9   22     3
29          10   23     3
30          10   23     1
31           8   19     2
32           9   19     2
```

2.1.1 Análisis mediante class

Como en el ejercicio uno, realizamos el análisis mediante class

```
> clasificacion = rpart (curso~., data = muestra, method ="class",minsplit=1)
> clasificacion

n= 32

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 32 22 1 (0.31250000 0.25000000 0.25000000 0.18750000)
   2) edad>=20.5 16  8 1 (0.50000000 0.06250000 0.31250000 0.12500000)
     4) edad< 21.5 3  0 1 (1.00000000 0.00000000 0.00000000 0.00000000) *
     5) edad>=21.5 13  8 1 (0.38461538 0.07692308 0.38461538 0.15384615)
      10) asignaturas>=9.5 5  2 1 (0.60000000 0.00000000 0.40000000 0.00000000) *
      11) asignaturas< 9.5 8  5 3 (0.25000000 0.12500000 0.37500000 0.25000000)
        22) edad>=22.5 5  3 1 (0.40000000 0.00000000 0.20000000 0.40000000)
          44) asignaturas< 8.5 3  1 1 (0.66666667 0.00000000 0.33333333 0.00000000) *
          45) asignaturas>=8.5 2  0 4 (0.00000000 0.00000000 0.00000000 1.00000000) *
        23) edad< 22.5 3  1 3 (0.00000000 0.33333333 0.66666667 0.00000000) *
   3) edad< 20.5 16  9 2 (0.12500000 0.43750000 0.18750000 0.25000000)
     6) asignaturas< 9.5 13  6 2 (0.07692308 0.53846154 0.23076923 0.15384615)
      12) edad< 19.5 8  3 2 (0.12500000 0.62500000 0.25000000 0.00000000) *
      13) edad>=19.5 5  3 2 (0.00000000 0.40000000 0.20000000 0.40000000)
        26) asignaturas>=8.5 4  2 2 (0.00000000 0.50000000 0.00000000 0.50000000) *
        27) asignaturas< 8.5 1  0 3 (0.00000000 0.00000000 1.00000000 0.00000000) *
     7) asignaturas>=9.5 3  1 4 (0.33333333 0.00000000 0.00000000 0.66666667) *

> rpart.plot(clasificacion)
```



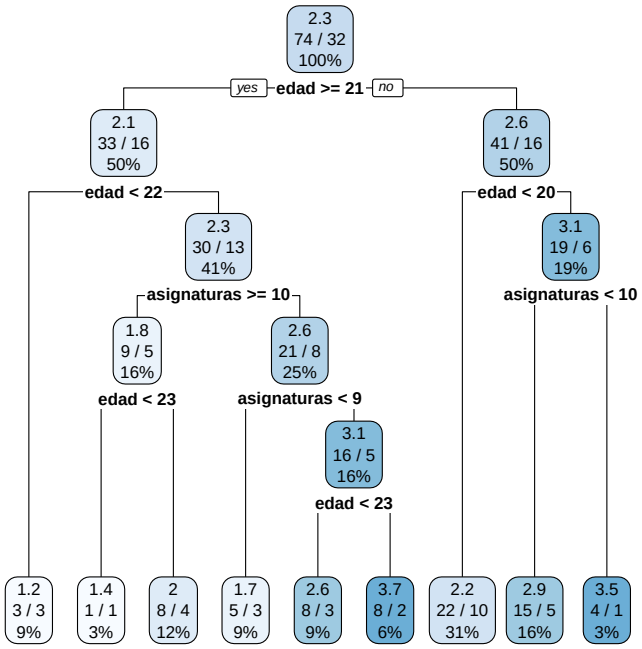
2.1.2 Análisis mediante Poisson

Este análisis realiza de acuerdo a la distribución de Poisson, que se trata de la probabilidad de que ocurra cierto suceso a partir de su frecuencia media. Esto se ve reflejado en el resultado dado, en el que aparecen las frecuencias absolutas y el resultado de la probabilidad del suceso por cada nodo.

```
> clasificacion = rpart (curso~., data = muestra, method ="poisson",minsplit=1)
> clasificacion

n= 32

node), split, n, deviance, yval
      * denotes terminal node

 1) root 32 17.38351000 2.312500
   2) edad>=20.5 16 10.13336000 2.069079
     4) edad< 21.5 3  0.07397486 1.165354 *
     5) edad>=21.5 13  7.73678600 2.307847
      10) asignaturas>=9.5 5  2.60774200 1.840796
        20) edad< 22.5 1  0.12490650 1.396226 *
        21) edad>=22.5 4  2.09483300 2.030488 *
      11) asignaturas< 9.5 8  4.19570800 2.608974
        22) asignaturas< 8.5 3  1.49496000 1.748031 *
        23) asignaturas>=8.5 5  0.92373510 3.129353
          46) edad< 22.5 3  0.26493320 2.622047 *
          47) edad>=22.5 2  0.04738466 3.700000 *
```

```
   3) edad< 20.5 16  6.38419800 2.555921
     6) edad< 19.5 10  3.44462100 2.204663 *
     7) edad>=19.5 6  1.61241900 3.109244
      14) asignaturas< 9.5 5  1.36424500 2.945274 *
      15) asignaturas>=9.5 1  0.07097567 3.490566 *
```

```
> rpart.plot(clasificacion)
```



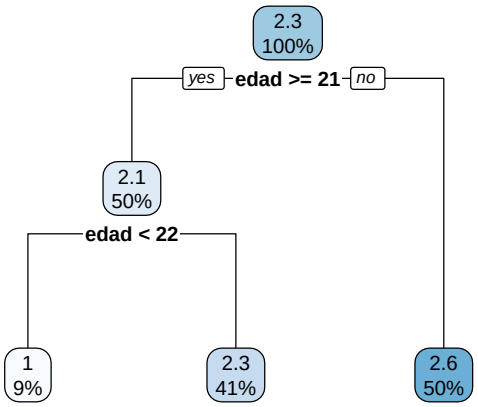
2.1.3 Análisis mediante Anova

El funcionamiento básico de anova consiste en el cálculo de la media de un conjunto de los valores como grupos para luego realizar una comparación de las varianzas de estas medias. En nuestro caso, respecto al árbol generado, el primer valor consiste en la media generada de cada grupo de acuerdo al elemento de su curso, y el segundo valor la probabilidad de este suceso en dicho nodo, desglosando el árbol y sus hijos de acuerdo a una agrupación por edad.

```
> clasificacion = rpart (curso~., data = muestra, method ="anova",minsplit=1)
> clasificacion

n= 32

node), split, n, deviance, yval
      * denotes terminal node

1) root 32 38.87500 2.312500
  2) edad>=20.5 16 20.93750 2.062500
```

```
    4) edad< 21.5 3  0.00000 1.000000 *
    5) edad>=21.5 13 16.76923 2.307692 *
  3) edad< 20.5 16 15.93750 2.562500 *
```

```
> rpart.plot(clasificacion)
```



2.1.4 Análisis mediante C50

C50 utiliza como medida de pureza, a la hora de generar las divisiones de los árboles, la entropía, además realiza poda del árbol de forma automática en caso de ser necesario.

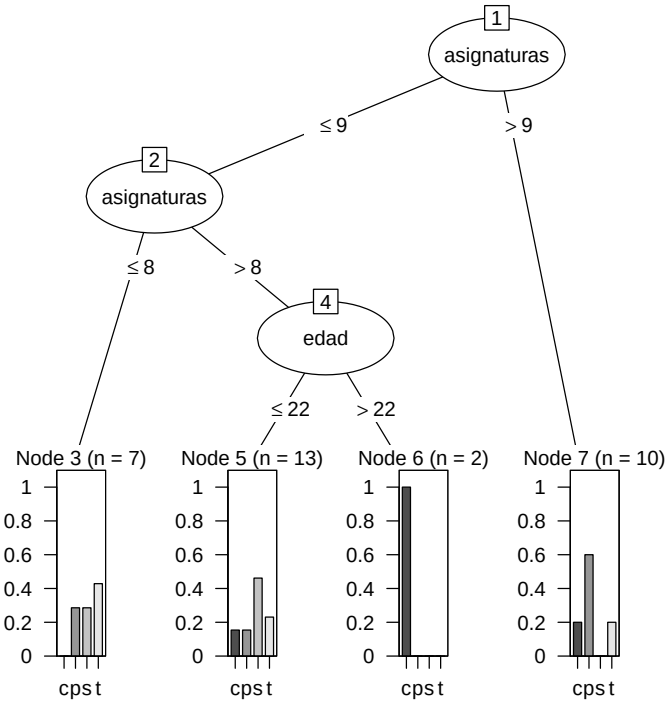
Para llevar a cabo esta ejecución utilizaremos nuevamente nuestro conjunto de datos de estudiantes, pero esta vez modificaremos los valores numéricos de la columna *curso* por valores cualitativos en forma de cadena que los representen.

```
> alu <- vector(length = dim(muestra)[1])
> alu[muestra$curso=="1"] <-"p"
> alu[muestra$curso=="2"] <-"s"
> alu[muestra$curso=="3"] <-"t"
> alu[muestra$curso=="4"] <-"c"
> muestra$curso <- factor(alu)
```

Con nuestra muestra modificada solo restaría ejecutar el algoritmo y mostrar sus resultados.

```
> modelo = C5.0(curso~.,data = muestra)
> summary(modelo)
```

```
Call:
C5.0.formula(formula = curso ~ ., data = muestra)


C5.0 [Release 2.07 GPL Edition]          Tue Dec 01 20:25:47 2020
-------------------------------

Class specified by attribute `outcome'

Read 32 cases (3 attributes) from undefined.data

Decision tree:

asignaturas > 9: p (10/4)
asignaturas <= 9:
:...asignaturas <= 8: t (7/4)
    asignaturas > 8:
    :...edad <= 22: s (13/7)
        edad > 22: c (2)


Evaluation on training data (32 cases):

           Decision Tree
         ----------------
         Size      Errors

            4   15(46.9%)   <<


          (a)   (b)   (c)   (d)    <-classified as
         ----  ----  ----  ----
            2     2     2          (a): class c
                  6     2     2    (b): class p
                        6     2    (c): class s
                  2     3     3    (d): class t


        Attribute usage:

        100.00%        asignaturas
         46.88%        edad


Time: 0.0 secs

> plot(modelo)
```



2.2 Apartado 2.2

Desarrollo por parte de cada alumno del enunciado y la solución de un ejercicio en el que se realice un análisis con R de clasificación supervisada con regresión introduciendo modificaciones sobre el ejercicio hecho en clase

En el ejercicio 1 realizamos un análisis de regresión de una muestra de datos pequeña; ahora utilizaremos la muestra de los estudiantes y comprobaremos su regresión, junto con su correlación.

```
> install.packages("tidyverse",repos = "https://cran.rediris.es/")
> library(ggpubr)
```

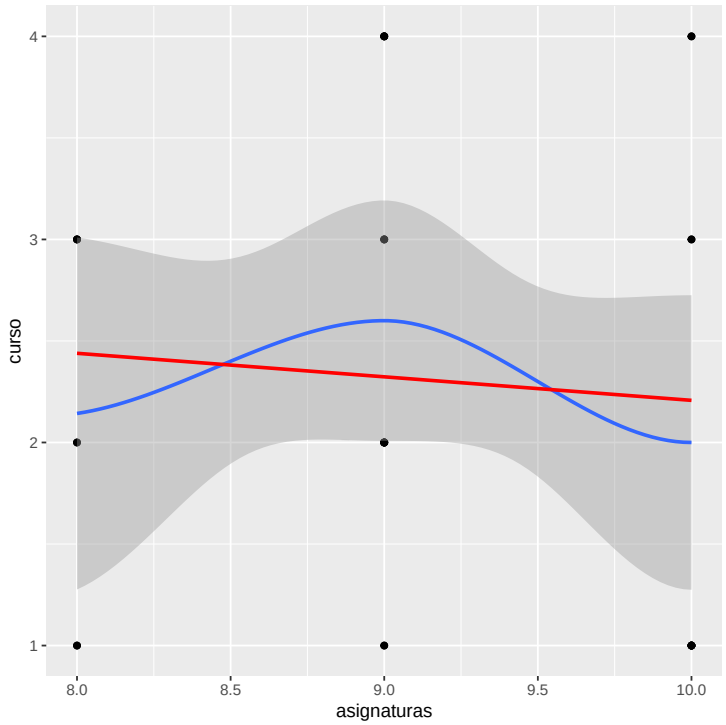
Vamos a estudiar la correlación entre las asignaturas de los alumnos y sus cursos, pero esta vez, además de realizar la salida de forma gráfica usando lm, es decir la correlación lineal (rojo), utilizaremos la función *stat-smooth* que nos muestra la correlación no lineal(azul) entre los valores.

```
> alumnos = read.table("datos.txt")
> alumnos
```

```
   asignaturas edad curso
1           10   20     4
2            8   23     3
3           10   19     4
4            8   20     3
5            8   19     3
6            9   21     1
```

```
7           10   19     1
8            9   19     1
9            9   22     2
10           9   19     3
11          10   22     1
12           9   19     2
13           9   20     2
14          10   21     1
15           8   23     1
16           9   20     4
17           9   23     4
18           9   22     3
19          10   23     3
20          10   23     1
21           8   19     2
22           9   19     2
23           9   20     2
24          10   21     1
25           8   23     1
26           9   20     4
27           9   23     4
28           9   22     3
29          10   23     3
30          10   23     1
31           8   19     2
32           9   19     2
```

```
> ggplot(alumnos, aes(x = asignaturas, y = curso)) +
+   geom_point() +
+   stat_smooth() +
+   stat_smooth(colour="red", method = lm, se= FALSE)
```



Además, vamos a analizar esto respecto de la correlación de los valores.

```
> cor(alumnos$asignaturas, alumnos$curso)
```

```
[1] -0.07599843
```

El resultado de la correlación es prácticamente 0, es decir muy malo y por lo tanto se puede entender que, en la regresión resultante, tal y como se observa en la gráfica mostrada, tampoco será buena. Debido a estas situaciones puede ser interesante la inclusión de la regresión no lineal para observar la variación de la regresión respecto de los datos.