PEC 3

Amelia Martínez Sequera, Sergio García Muñoz

20/5/2020

El archivo Rmd de este documento y los datos que se han utilizado se encuentran en: https://github.com/gititub/Shiny-app

Paquetes necesarios para la ejecución de este archivo:

```
library(tables)
library(magrittr)
library(kableExtra)
library(sqldf)
library(Hmisc)
library(readr)
library(ggplot2)
library(lattice)
library(colorspace)
library(fastGraph)
library(dplyr)
library(tidyr)
library(car)
library(lmtest)
library(foreign)
library(MVN)
library(corrplot)
library(RColorBrewer)
library(psych)
library(lmtest)
library(grid)
library(vcd)
```

SECCIÓN 1 (8 puntos)

(1) (0.5p)

Hemos seleccionado el conjunto de datos titulado “Insulin Secretion of Asian Indians. Studying the relative contribution of Insulin Resistance & Secretion in Diabetes”. Está disponible en el sitio web de Kaggle, una comunidad online dedicada a la ciencia de datos y el machine learning. Los datos pueden consultarse o descargarse desde la dirección:

https://www.kaggle.com/jit1806/insulin-secretion-of-asian-indians

Podemos decir que los motivos que nos han llevado a escoger estos datos son:

- Son datos reales de una población concreta y bien definida. Pertenecen a voluntarios reclutados a partir de un programa de screening de salud “From Food to Nutrition” impulsado por la asociación sin ánimo de lucro SWANIRVAR, entre enero de 2017 y septiembre de 2018.
- La población es bastante numerosa, lo que esperamos facilite la obtención de resultados con una significación adecuada.
- Recogen una gran cantidad de variables, clínicas, bioquímicas y demográficas, por lo que

pensamos que de ellos se puede extraer bastante información.

(2) (0.5p)

Se trata de un fichero de datos csv (comma separated values). Utilizamos un read.csv para importarlo como dataframe. Hay que añadir el parámetro skipNul=TRUE, ya que contiene embeded null characters.

```
url<-"https://github.com/gititub/Shiny-app/blob/master/sage_tae_2019_df.csv"
datosdf<- read.csv("sage_tae_2019_df.csv", skipNul = T)
```

El fichero contiene 23 variables:

```
head(datosdf)
##   þÿ AGE SEX      BMI  WC FBS..mmol.l.        TG       TC SBP DBP
DM_status
## 1  1  47   M 24.30194 101     4.290157 199.08320 145.9839  NA  NA
Healthy
## 2  2  57   F 23.71338  81     4.761028  69.77708 183.5294  NA  NA
Healthy
## 3  3  46   M 19.92188  79     5.287487 115.59184 214.7174  NA  NA
Healthy
## 4  4  51   M 25.84312  95     4.842776 219.12716 207.6401  NA  NA
Healthy
## 5  5  42   M 21.82657  89     5.117451 267.35322 213.3979  NA  NA
Healthy
## 6  6  38   F 36.27000 115     5.982906 187.00000 230.0000  NA  NA
Healthy
##   F.Ins.pmol.L. Adiponectin.microgm.ml...5000.X. Leptin..ng.ml. HOMA2..B
## 1      33.28106                              NA             NA     94.3
## 2      29.64733                              NA             NA     70.7
## 3      22.01315                        1.654843             NA     46.8
## 4      33.58116                              NA             NA     74.2
## 5      62.79710                              NA             NA    101.3
## 6     115.68785                        1.772031         168.78    113.5
##   HOMA2..S  HOMA2.IR Body.Fat....from.Age..BMI BMI_group       FBS
Ins
## 1    166.6 0.6002401                  19.06290      Lean  77.22282
4.792089
## 2    181.5 0.5509642                  31.48007      Lean  85.69850
4.268874
## 3    237.3 0.4214075                  12.36281      Lean  95.17477
3.169640
## 4    159.6 0.6265664                  21.89468     Obese  87.16997
4.835300
## 5     84.8 1.1792453                  14.69986      Lean  92.11411
9.042058
## 6     45.0 2.2222222                  47.84500     Obese 107.69231
16.657718
##     HOMA_IR    HOMA_B
## 1 0.9137250 121.29463
## 2 0.9032991  67.70469
## 3 0.7448636  35.46475
## 4 1.0407233  72.01945
## 5 2.0565462 111.80629
## 6 4.4294027 134.17921
str(datosdf)
```

```
## 'data.frame':    650 obs. of  23 variables:
##  $ þÿ                               : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ AGE                              : int  47 57 46 51 42 38 55 43 45 36
...
##  $ SEX                              : chr  "M" "F" "M" "M" ...
##  $ BMI                              : num  24.3 23.7 19.9 25.8 21.8 ...
##  $ WC                               : num  101 81 79 95 89 115 100 NA 86
129 ...
##  $ FBS..mmol.l.                     : num  4.29 4.76 5.29 4.84 5.12 ...
##  $ TG                               : num  199.1 69.8 115.6 219.1 267.4 ...
##  $ TC                               : num  146 184 215 208 213 ...
##  $ SBP                              : int  NA NA NA NA NA NA NA NA NA NA
...
##  $ DBP                              : int  NA NA NA NA NA NA NA NA NA NA
...
##  $ DM_status                        : chr  "Healthy" "Healthy" "Healthy"
"Healthy" ...
##  $ F.Ins.pmol.L.                    : num  33.3 29.6 22 33.6 62.8 ...
##  $ Adiponectin.microgm.ml...5000.X.: num  NA NA 1.65 NA NA ...
##  $ Leptin..ng.ml.                   : num  NA NA NA NA NA ...
##  $ HOMA2..B                         : num  94.3 70.7 46.8 74.2 101.3 ...
##  $ HOMA2..S                         : num  166.6 181.5 237.3 159.6 84.8 ...
##  $ HOMA2.IR                         : num  0.6 0.551 0.421 0.627 1.179 ...
##  $ Body.Fat....from.Age..BMI        : num  19.1 31.5 12.4 21.9 14.7 ...
##  $ BMI_group                        : chr  "Lean" "Lean" "Lean" "Obese" ...
##  $ FBS                              : num  77.2 85.7 95.2 87.2 92.1 ...
##  $ Ins                              : num  4.79 4.27 3.17 4.84 9.04 ...
##  $ HOMA_IR                          : num  0.914 0.903 0.745 1.041 2.057
...
##  $ HOMA_B                           : num  121.3 67.7 35.5 72 111.8 ...
datosdf$SEX<-as.factor(datosdf$SEX)
datosdf$DM_status<-as.factor(datosdf$DM_status)
datosdf$BMI_group<-as.factor(datosdf$BMI_group)
```

Como puede verse, hay 650 observaciones de 23 variables, la mayoría numéricas, habiendo tres que por ser categóricas convertimos en factores, y la primera que es un contador. A continuación hacemos una pequeña descripción de las mismas, así como su significado:

Variables demográficas y antropométricas:

AGE: Edad

SEX: Sexo

BMI: índice de masa corporal

WC: Perímetro abdominal

Body.Fat….from.Age..BMI: Porcentaje de grasa corporal calculado a partir de la edad y el BMI.

BMI_group: Factor categorizando el BMI

Variables bioquímicas y fisiológicas relacionadas con el perfil glucémico, lipídico y el riesgo cardiovascular:

FBS: Glucosa plasmática en ayunas (mmol/L)

TG: Triglicéridos (mg/dL)

TC: Colesterol total (mg/dL)

SBP: Presión arterial sistólica (mmHg)

DBP: Presión arterial diastólica (mmHg)

Variable que define el estado de presencia o ausencia de diabetes:

DM_status: Categórica con dos posibles valores: Healthy y NDM, entendido este último como diagnóstico de Diabetes tipo 2 reciente.

Variables relacionadas con el estado metabólico y la resistencia a la insulina:

F.Ins.pmol.L: Concentración plasmática en ayunas de insulina (pmol/L).

Adiponectin.microgm.ml…5000.X: Concentración de adiponectina (ug/mL). Se trata de una hormona secretada por los adipocitos, que participa en el metabolismo de la glucosa y los ácidos grasos.

Leptin..ng.ml.: Concentración de leptina (ng/mL). Se trata de una hormona, una adipoquina secretada mayoritariamente por los adipocitos y ejerce un feedback negativo sobre el hipotálamo para inhibir el apetito.

HOMA2..B, HOMA2..S y HOMA2.IR: Índices HOMA (Homeostaic Model Assessment). Se trata de índices calculados a partir de los niveles de insulina y glucosa que tratan de medir el grado de resistencia a la insulina. Se han calculado tres índices, el de función de células beta pancreáticas (HOMA-B), el de sensibilidad (HOMA-S) y el de resistencia a la insulina (HOMA-IR).

El resto de variables son iguales a otras ya descritas pero expresadas en unidades convencionales en lugar de unidades SI.

(3) (1.5 p)

P1. ¿Qué valores medios presentan las variables bioquímicas y fisiológicas según presencia de diabetes o no, categoría de índice de masa corporal y sexo? Presentar en formato tabla calculando sus intervalos de confianza.

Utilizamos la función tabular() contenida en el paquete tables, ya que permite de una manera sencilla crear tablas con selecciones de datos y aplicando funciones. Para hallar la media, creamos una función “Media” con el argumento na.rm=TRUE, para que no tenga en cuenta los valores NA presentes en algunos datos. Para calcular los intervalos de confianza creamos otra función “IC” que usa t.test para calcularlos y devuelve el resultado en forma de cadena de valores separados por un guión. Tanto a la función “Media” como a “IC” se les aplica un round para limitar el número de decimales. Por último, creamos la tabla con ayuda de la librería kableExtra y le damos formato.

```
attach(datosdf)
Media <- function(x){
  round(mean(x, na.rm=TRUE), 1)
  }
IC <- function(x){
 int<-c(t.test(x)$conf.int)
  result<-paste(round(int[1],1),round(int[2],1), sep="-")
```

```
  return(result)
}
biofis<- tabular ((DM_status*BMI_group*SEX) ~ (FBS+TG+TC+SBP+DBP)*(Media+IC))
biofis_kable<-toKable(biofis)
kable_styling(biofis_kable, full_width = TRUE, bootstrap_options="striped",
row_label_position="c")
```

P2. ¿Cuántos individuos hay que sean diabéticos y de sexo femenino, y que tengan una concentración de glucosa plasmática por encima de 200 mg/dL, y y qué porcentaje representan respecto al total de individuos de sexo femenino? ¿Y si ahora consideramos el sexo masculino? ¿Cuáles son las concentraciones mínima y máxima de glucosa en todos los pacientes? ¿Qué porcentaje de pacientes obesos tienen niveles de insulina elevados si consideramos un nivel de insulina elevado aquel que sea superior al percentil 95 de los individuos sanos?

Podemos hacer este tipo de consultas usando SQL. Para ello, cargamos las librerías sqldf y usamos SELECT con WHERE e IF. Es importante tener en cuenta que sqldf trabaja con dataframes y que cualquier variable que participe debe estar dentro del dataframe, o bien incorporarse como un dataframe en sí. Para responder a la última pregunta hemos tenido que convertir el percentil 95 de la concentración de insulina en personas sanas (In_S95) en un data frame, para poder incluirlo en el SELECT correspondiente que selecciona los valores de insulina en pacientes obesos por encima de este percentil.

```
#Seleccionamos el sexo de los casos con glucosa>200, diabetes y sexo
femenino, y usamos un nrow para contar el número de casos:
DM_SF<-nrow(sqldf("SELECT SEX FROM datosdf WHERE FBS>200 AND DM_status='NDM'
AND SEX='F'"))
DM_SFp<-DM_SF/nrow(sqldf("SELECT FBS FROM datosdf WHERE SEX='F' AND
DM_status='NDM'"))*100

paste("El número de mujeres diabéticas con glucosa > 200 mg/dL es:",DM_SF,"y
representan un",round(DM_SFp, 1),"% del total de mujeres diabéticas")
## [1] "El número de mujeres diabéticas con glucosa > 200 mg/dL es: 34 y
representan un 33.7 % del total de mujeres diabéticas"
#Hacemos lo mismo con el sexo masculino:
DM_SM<-nrow(sqldf("SELECT SEX FROM datosdf WHERE FBS>200 AND DM_status='NDM'
AND SEX='M'"))
DM_SMp<-DM_SM/nrow(sqldf("SELECT FBS FROM datosdf WHERE SEX='M' AND
DM_status='NDM'"))*100

paste("El número de hombres diabéticos con glucosa > 200 mg/dL es:",DM_SM,"y
representan un",round(DM_SMp, 1),"% del total de hombres diabéticos")
## [1] "El número de hombres diabéticos con glucosa > 200 mg/dL es: 26 y
representan un 35.1 % del total de hombres diabéticos"
paste("La concentración máxima de glucosa es:", round(max(FBS), 1),
"mg/dL","y la mínima: ", round(min(FBS), 1), "mg/dL")
## [1] "La concentración máxima de glucosa es: 379.1 mg/dL y la mínima:  66
mg/dL"
#Seleccionamos niveles de insulina de sujetos sanos y calculamos el percentil
95. Posteriormente lo usamos para seleccionar los obsesos con niveles
```

```
superiores a este percentil:
In_S<-sqldf("SELECT Ins FROM datosdf WHERE DM_status='Healthy'")
In_S95<-quantile(In_S[,1], probs = 0.95)
In_S95<-as.data.frame(In_S95)
In_Ob<-fn$sqldf("SELECT Ins FROM datosdf, In_S95 WHERE BMI_group='Obese' AND
Ins > In_S95")
In_Obp<-nrow(In_Ob)/nrow(sqldf("SELECT Ins FROM datosdf WHERE
BMI_group='Obese'"))*100

paste("El porcentaje de obesos con niveles de insulina superiores al
percentil 95 de los niveles de pacientes sanos es:", round(In_Obp, 1),"%")
## [1] "El porcentaje de obesos con niveles de insulina superiores al
percentil 95 de los niveles de pacientes sanos es: 16.7 %"
```

P3. El síndrome metabólico se define como un conjunto de factores que conllevan un aumento del riesgo de padecer una enfermedad cardiovascular o diabetes mellitus tipo 2. La European Group for the Study of Insulin Resistance lo define mediante los siguientes criterios: 1. presencia del fenómeno de resistencia a la insulina (incremento del 25% de los valores de insulina en ayunas entre los individuos no-diabéticos) 2. Presencia de dos o más de los siguientes factores: - Obesidad central: diámetro de cintura ≥ 94 cm (en hombres), ≥ 80 cm (en mujeres). - Dislipidemia: TG ≥ 177 mg/dL y/o HDL-C < 1.0 mg/dL, o ser tratado por dislipidemia. - Hipertensión arterial: ≥ 140/90 mmHg o estar bajo tratamiento antihipertensivo - Hemoglobina glicosilada ≥ 6.1 mmol/L.

En base a los datos disponibles, cuántos individuos no serían diabéticos pero padecerían síndrome metabólico?

En primer lugar calculamos la insulina media de los sujetos sanos, después seleccionamos el grupo de interés (sanos, excluyendo diabéticos) mediante un SELECT. Por último creamos una función que nos permita evaluar este o cualquier otro grupo de pacientes, que contiene un IF con todas las condiciones del síndrome metabólico:

```
#Calculamos la media de los valores de insulina entre los sujetos sanos:
In_Sm<-mean(In_S[,1])
#Seleccionamos los individuos sanos:
healthy<-sqldf("SELECT * FROM datosdf WHERE DM_Status='Healthy'")
healthy<-na.omit(healthy)
#Creamos la función que crea un dataframe con todos los datos de los casos
que cumplen criterios de síndrome metabólico:
MS<-function(x) {
  res<-c()
  for(i in 1:nrow(x)){
        if(

(x$Ins[i]>1.25*In_Sm
& ((x$SEX[i]=="F" & x$WC[i]>=80) || (x$SEX[i]=="M" & x$WC[i]>=94))
& x$TG[i]>=177)||

(x$Ins[i]>1.25*In_Sm
& ((x$SEX[i]=="F" & x$WC[i]>=80) || (x$SEX[i]=="M" & x$WC[i]>=94))
& x$SBP[i]>=140 & x$DBP[i]>=90)||

(x$Ins[i]>1.25*In_Sm
& x$TG[i]>=177 & x$SBP[i]>=140 & x$DBP[i]>=90)
```

```
)
{res<-rbind(res, x[i,])}
}
  return(res)
}

Heal_MS<-MS(healthy)
paste("El número de individuos no diabéticos que padecen síndrome metabólico
es de:", nrow(Heal_MS))
## [1] "El número de individuos no diabéticos que padecen síndrome metabólico
es de: 17"
```

P4. ¿Cuáles son los valores medios de las variables bioquímicas relacionadas con el metabolismo carbohidrato y lipídico, así como las relacionadas con la resistencia a la insulina en el grupo de pacientes diabéticos? ¿Y en los no diabéticos con síndrome metabólico? ¿Y en los sanos (no diabéticos y sin síndrome metabólico)? Acompañar las medias de sus correspondientes IC.

Nuevamente usamos SQL y después un loop para construir un dataframe con la media y el IC de aquellos casos que cumplen lascondiciones requeridas. Después construimos tablas con kable. Para el caso de los sanos hay que introducir un paso adicional, que ha consistido en crear un subset usando la condición de que NO se cumplan los criterios de síndrome metabólico:

```
bioq_DM<- sqldf("SELECT FBS, TG, TC, Ins, HOMA_B, HOMA_IR, `HOMA2..B`,
`HOMA2..S`, `HOMA2.IR`, `Adiponectin.microgm.ml...5000.X.`, `Leptin..ng.ml.`
FROM datosdf WHERE DM_status=='NDM'")
bioq_DM_m<-c()
for(i in 1:ncol(bioq_DM)) {
    m<-Media(bioq_DM[,i])
    ic<-IC(bioq_DM[,i])
    bioq_DM_m<-cbind(bioq_DM_m, m, ic)
}
cnames<-c("Glucosa", "IC", "Triglicéridos", "IC", "Colesterol",
"IC","Insulina", "IC", "HOMA_B", "IC", "HOMA_IR", "IC", "HOMA2_B", "IC",
"HOMA2_S", "IC", "HOMA2_IR", "IC", "Adiponectina", "IC", "Leptina", "IC")
colnames(bioq_DM_m)<-cnames
kable(bioq_DM_m, align = "c", padding=5, format = "pandoc", caption = "Medias
parámetros bioquímicos y de resistencia a insulina en individuos diabéticos")
```

Medias parámetros bioquímicos y de resistencia a insulina en individuos diabéticos

Glucosa	IC	Triglicéridos	IC	Colesterol	IC	Insulina	IC	HOMA_B	IC	HOMA_IR	IC	HOMA2_B	IC	HOMA2_S	IC	HOMA2_IR	IC	Adiponectina	IC	Leptina	IC
187.1	178.3-196	162.3	146.8-177.8	181.7	172.4-191	7.4	6.3-8.4	26.7	22.3-31.2	3.3	2.8-3.8	26.9	23.3-30.5	178.2	156.8-199.6	1.2	1-1.3	4.3	3.6-5	29.9	20.7-39

```
bioq_HeMS<- sqldf("SELECT FBS, TG, TC, Ins, HOMA_B, HOMA_IR, `HOMA2..B`,
`HOMA2..S`, `HOMA2.IR`, `Adiponectin.microgm.ml...5000.X.`, `Leptin..ng.ml.`
FROM Heal_MS")
```

```
bioq_HeMS_m<-c()
for(i in 1:ncol(bioq_HeMS)) {
    m<-Media(bioq_HeMS[,i])
    ic<-IC(bioq_HeMS[,i])
    bioq_HeMS_m<-cbind(bioq_HeMS_m, m, ic)
}
cnames<-c("Glucosa", "IC", "Triglicéridos", "IC", "Colesterol",
"IC","Insulina", "IC", "HOMA_B", "IC", "HOMA_IR", "IC", "HOMA2_B", "IC",
"HOMA2_S", "IC", "HOMA2_IR", "IC", "Adiponectina", "IC", "Leptina", "IC")
colnames(bioq_HeMS_m)<-cnames
kable(bioq_HeMS_m, align = "c", padding=5, format = "pandoc", caption =
"Medias parámetros bioquímicos y de resistencia a insulina en individuos no
diabéticos con síndrome metabolico")
```

Medias parámetros bioquímicos y de resistencia a insulina en individuos no diabéticos con síndrome metabolico

Glucosa	IC	Triglicéridos	IC	Colesterol	IC	Insulina	IC	HOMA_B	IC	HOMA_IR	IC	HOMA2_B	IC	HOMA2_S	IC	HOMA2_IR	IC	Adiponectina	IC	Leptina	IC
92.9	87.3-98.5	169.3	121.7-216.9	166.4	149-183.8	8.6	7.5-9	117.7	90.3-145.2	2	1.7-2.3	99.3	85.5-113.1	93.8	82.9-104.7	1.1	1-1.3	4.1	2.7-5.4	45	25.7-64.4

```
#Definimos un subset del dataframe healthy que contiene todos los individuos
sin diabetes aplicando la condición de que NO cumplan lo requisitos para
tener síndrome metabólico. Posteriormente continuamos como en los apartados
anteriores:

Heal_NoMS<-subset.data.frame(healthy, subset =! (Ins>1.25*In_Sm
                        & ((SEX=="F" & WC>=80) || (SEX=="M" & WC>=94)&
TG>=177)||
                                                    (Ins>1.25*In_Sm
                & ((SEX=="F" & WC>=80) || (SEX=="M" & WC>=94)) & SBP>=140 &
DBP>=90)||
                                                   (Ins>1.25*In_Sm
                        & TG>=177 & SBP>=140 & DBP>=90)))

bioq_HeNoMS<- sqldf("SELECT FBS, TG, TC, Ins, HOMA_B, HOMA_IR, `HOMA2..B`,
`HOMA2..S`, `HOMA2.IR`, `Adiponectin.microgm.ml...5000.X.`, `Leptin..ng.ml.`
FROM Heal_NoMS")

bioq_HeNoMS_m<-c()
for(i in 1:ncol(bioq_HeNoMS)) {
    m<-Media(bioq_HeNoMS[,i])
    ic<-IC(bioq_HeNoMS[,i])
    bioq_HeNoMS_m<-cbind(bioq_HeNoMS_m, m, ic)
}
cnames<-c("Glucosa", "IC", "Triglicéridos", "IC", "Colesterol",
"IC","Insulina", "IC", "HOMA_B", "IC", "HOMA_IR", "IC", "HOMA2_B", "IC",
"HOMA2_S", "IC", "HOMA2_IR", "IC", "Adiponectina", "IC", "Leptina", "IC")
```

```
colnames(bioq_HeNoMS_m)<-cnames
kable(bioq_HeNoMS_m, align = "c", padding=5, format = "pandoc", caption =
"Medias parámetros bioquímicos y de resistencia a insulina en individuos no
diabéticos sin síndrome metabolico")
```

Medias parámetros bioquímicos y de resistencia a insulina en individuos no diabéticos sin síndrome metabolico

Glucosa	IC	Triglicéridos	IC	Colesterol	IC	Insulina	IC	HOMA_B	IC	HOMA_IR	IC	HOMA2_B	IC	HOMA2_S	IC	HOMA2_IR	IC	Adiponectina	IC	Leptina	IC
84.8	83.5-86.2	103.7	91.7-115.7	161.8	156.5-167	4.3	3.9-4.7	80.1	73.8-86.9	0.9	0.8-1	69.9	66.2-73.7	269.4	248.1-290.7	0.6	0.5-0.6	4	3.6-4.4	27.6	24-31.1

P5. ¿Cuáles son los valores medios de las variables bioquímicas relacionadas con el metabolismo carbohidrato y lipídico, así como las relacionadas con la resistencia a la insulina en el grupo de pacientes obesos y diabéticos? ¿Y en el de los obesos no diabéticos pero con síndrome metabólico? ¿Entre los individuos obesos del estudio, qué proporción son diabéticos, qué proporción tienen síndrome metabólico sin ser diabéticos y qué proporción son sanos? Acompañar las medias de sus correspondientes IC.

```
# Operamos como en la pregunta 4 pero añadiendo la condición de que los
sujetos sean obesos:
bioq_DM_Ob<- sqldf("SELECT FBS, TG, TC, Ins, HOMA_B, HOMA_IR, `HOMA2..B`,
`HOMA2..S`, `HOMA2.IR`, `Adiponectin.microgm.ml...5000.X.`, `Leptin..ng.ml.`
FROM datosdf WHERE DM_status=='NDM' AND BMI_group=='Obese'")
bioq_DM_Ob_m<-c()
for(i in 1:ncol(bioq_DM_Ob)) {
    m<-Media(bioq_DM_Ob[,i])
    ic<-IC(bioq_DM_Ob[,i])
    bioq_DM_Ob_m<-cbind(bioq_DM_Ob_m, m, ic)
}
cnames<-c("Glucosa", "IC", "Triglicéridos", "IC", "Colesterol",
"IC","Insulina", "IC", "HOMA_B", "IC", "HOMA_IR", "IC", "HOMA2_B", "IC",
"HOMA2_S", "IC", "HOMA2_IR", "IC", "Adiponectina", "IC", "Leptina", "IC")
colnames(bioq_DM_Ob_m)<-cnames
kable(bioq_DM_Ob_m, align = "c", padding=5, format = "pandoc", caption =
"Medias parámetros bioquímicos y de resistencia a insulina en individuos
obesos y diabéticos")
```

Medias parámetros bioquímicos y de resistencia a insulina en individuos obesos y diabéticos

Glucosa	IC	Triglicéridos	IC	Colesterol	IC	Insulina	IC	HOMA_B	IC	HOMA_IR	IC	HOMA2_B	IC	HOMA2_S	IC	HOMA2_IR	IC	Adiponectina	IC	Leptina	IC
176.9	165.3-188	154.6	137.8-175.7	185.8	170.3-201.3	9.3	7-12	36.2	28.7-45.7	4	3-5.3	34.9	29-42	138.3	109.4-167.2	1.4	1-1.9	4.7	2-9.4	58	37.2-78.8

Glucosa	IC	Triglicéridos	IC	Colesterol	IC	Insulina	IC	HOMA_B	IC	HOMA_IR	IC	HOMA2_B	IC	HOMA2_S	IC	HOMA2_IR	IC	Adiponectina	IC	Leptina	IC
8.5		17	1.5	20	1.4	10	.9	43.7		4	.7	0	.8	16	7.2	1	.6	5	.8	78.9	

```
#Para seleccionar pacientes no diabéticos con síndrome metabólico y obesos podemos usar el dataframe donde están recogidos todos los que tienen síndrome metabólico y seleccionar por BMI_group:
bioq_nDM_MS_Ob<- sqldf("SELECT FBS, TG, TC, Ins, HOMA_B, HOMA_IR, `HOMA2..B`, `HOMA2..S`, `HOMA2.IR`, `Adiponectin.microgm.ml...5000.X.`, `Leptin..ng.ml.` FROM Heal_MS WHERE BMI_group=='Obese'")
#Aplicamos un loop para calcular las medias e IC a cada varaible de la selección:
bioq_nDM_MS_Ob_m<-c()
for(i in 1:ncol(bioq_nDM_MS_Ob)) {
    m<-Media(bioq_nDM_MS_Ob[,i])
    ic<-IC(bioq_nDM_MS_Ob[,i])
    bioq_nDM_MS_Ob_m<-cbind(bioq_nDM_MS_Ob_m, m, ic)
}
cnames<-c("Glucosa", "IC", "Triglicéridos", "IC", "Colesterol", "IC","Insulina", "IC", "HOMA_B", "IC", "HOMA_IR", "IC", "HOMA2_B", "IC", "HOMA2_S", "IC", "HOMA2_IR", "IC", "Adiponectina", "IC", "Leptina", "IC")
colnames(bioq_nDM_MS_Ob_m)<-cnames
kable(bioq_nDM_MS_Ob_m, align = "c", padding=5, format = "pandoc", caption = "Medias parámetros bioquímicos y de resistencia a insulina en individuos obesos, NO diabéticos, CON síndrome metabólico")
```

Medias parámetros bioquímicos y de resistencia a insulina en individuos obesos, NO diabéticos, CON síndrome metabólico

Glucosa	IC	Triglicéridos	IC	Colesterol	IC	Insulina	IC	HOMA_B	IC	HOMA_IR	IC	HOMA2_B	IC	HOMA2_S	IC	HOMA2_IR	IC	Adiponectina	IC	Leptina	IC
96.9	87.2-106.7	126.7	71.7-181.3	150.1	126.9-173.3	8.1	.4-19	96.8	60.6-133	2	.4-3.7	87.3	68.6-106	97.8	80.6-115.1	1.1	.8-1.7	2.9	.6-5.1	52.9	14.1-91.7

```
#Para conocer la proporción de individuos obesos que tienen diabetes, los que tienen síndrome metabólico y los que son sanos, calculamos el número de obesos total y el de obesos diabéticos en datosdf, el número de obesos no diabéticos pero con síndrome metabólico en Heal_MS, y el de obesos sanos Heal_NoMS. Para ello hacemos subsets con las condiciones de interés y contamos filas. Hay que tener en cuenta que los porcentajes no sumarán 100 debido a los casos en los que no se ha podido determinar la presencia de síndrome metabólico por falta de datos.

pOb_SM<-nrow(subset.data.frame(Heal_MS, subset = (BMI_group=='Obese'))) /
```

```
nrow(subset.data.frame(datosdf, subset = (BMI_group=='Obese')))*100
pOb_DM<-nrow(subset.data.frame(datosdf, subset = (BMI_group=='Obese' &
DM_status=='NDM'))) / nrow(subset.data.frame(datosdf, subset =
(BMI_group=='Obese')))*100
pOb_H<-nrow(subset.data.frame(Heal_NoMS, subset = (BMI_group=='Obese'))) /
nrow(subset.data.frame(datosdf, subset = (BMI_group=='Obese')))*100

paste("Entre los individuos obesos, son diabéticos el",round(pOb_DM, 1),"%.")
## [1] "Entre los individuos obesos, son diabéticos el 31.4 %."
paste("Entre los individuos obesos, tienen síndrome metabólico
el",round(pOb_SM, 1),"%.")
## [1] "Entre los individuos obesos, tienen síndrome metabólico el 3.3 %."
paste("Entre los individuos obesos, son sanos el",round(pOb_H, 1),"%.")
## [1] "Entre los individuos obesos, son sanos el 24.5 %."
```

P6. El estándar de oro para medir la sensibilidad a la insulina corporal total se denomina clamp hiperinsulinémico-euglucémico, pero se trata de un procedimiento complejo y no se usa en la práctica. En su lugar se emplea el índice HOMA-IR. Se suele emplear un valor de 2,6 como punto de corte para diagnosticar la resistencia a la insulina, resultando útil, sobre todo, para evaluar dicha resistencia en pacientes con normoglucemia. Posteriormente, un modelo actualizado (HOMA2) añadió los efectos de la resistencia a la glucosa hepática y periférica, las variaciones que se producen en la secreción de insulina a concentraciones de glucosa elevadas, así como la contribución de la proinsulina circulante. ¿Cuáles son los valores medios y sus IC de HOMA-IR y HOMA2_IR en cada uno de los siguientes grupos?:

- No diabéticos, no obesos
- No diabéticos, obesos
- Diabéticos, no obesos
- Diabéticos, obesos

```
# Hacemos uso nuevamente de tabular y toKable:
homa_ir<- tabular ((DM_status*BMI_group) ~ (HOMA_IR+HOMA2.IR)*(Media+IC))
homa_ir_kable<-toKable(homa_ir)
kable_styling(homa_ir_kable, full_width = TRUE, bootstrap_options="striped",
row_label_position="c")
```

(4) (1.5p)

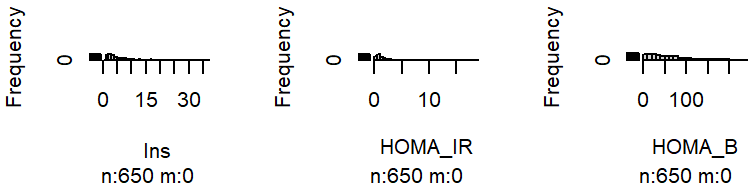
Para el resumen paramétrico usamos un summary:

```
summary(datosdf)
##        þÿ             AGE         SEX          BMI              WC
##  Min.   :  1.0   Min.   : 16.00   F:417   Min.   :13.22   Min.   : 54.00
##  1st Qu.:163.2   1st Qu.: 35.00   M:233   1st Qu.:20.78   1st Qu.: 78.00
##  Median :325.5   Median : 44.00           Median :23.81   Median : 86.00
##  Mean   :325.5   Mean   : 44.33           Mean   :24.36   Mean   : 85.91
##  3rd Qu.:487.8   3rd Qu.: 53.00           3rd Qu.:26.57   3rd Qu.: 94.00
##  Max.   :650.0   Max.   :104.00           Max.   :52.50   Max.   :143.00
##                  NA's   :1                                NA's   :91
##   FBS..mmol.l.          TG               TC              SBP
##  Min.   : 3.665   Min.   :  17.58   Min.   : 53.09   Min.   : 83.0
##  1st Qu.: 4.578   1st Qu.:  70.69   1st Qu.:130.80   1st Qu.:111.0
##  Median : 5.225   Median : 105.04   Median :162.11   Median :124.0
```

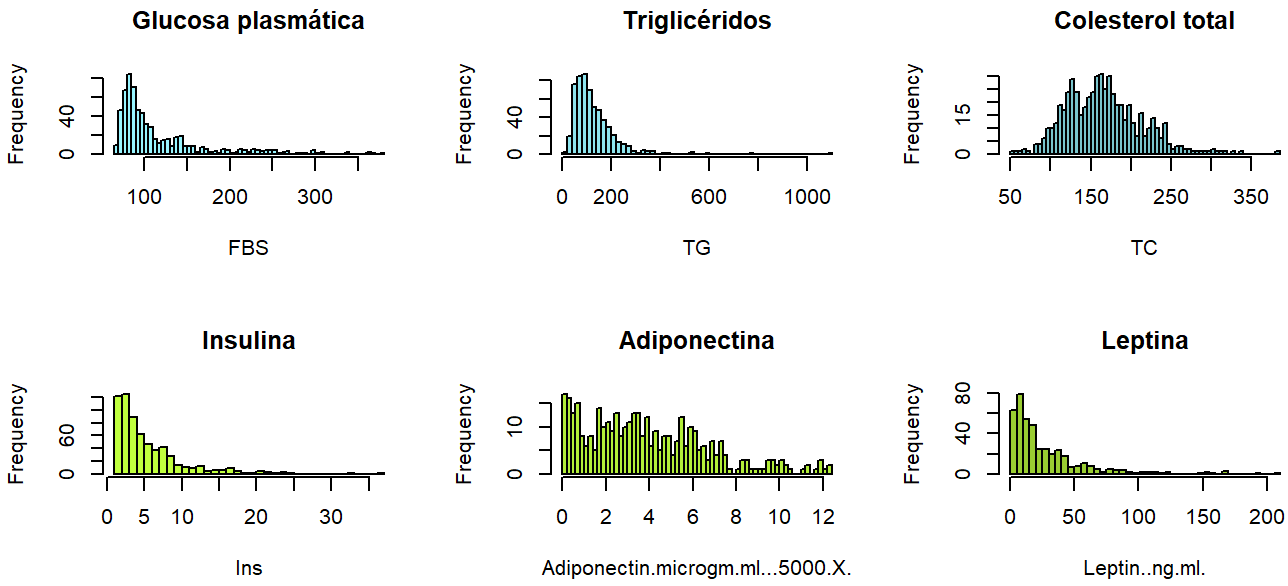
```
##  Mean   : 6.441   Mean   : 126.64   Mean   :166.57   Mean   :127.6
##  3rd Qu.: 7.159   3rd Qu.: 159.00   3rd Qu.:193.46   3rd Qu.:139.0
##  Max.   :21.062   Max.   :1084.24   Max.   :381.06   Max.   :198.0
##                   NA's   :73        NA's   :74       NA's   :193
##       DBP           DM_status   F.Ins.pmol.L.
##  Min.   : 29.00   Healthy:475   Min.   :  7.091
##  1st Qu.: 71.00   NDM    :175   1st Qu.: 16.285
##  Median : 80.00                 Median : 26.642
##  Mean   : 79.98                 Mean   : 38.723
##  3rd Qu.: 88.00                 3rd Qu.: 49.302
##  Max.   :121.00                 Max.   :251.722
##  NA's   :211
##  Adiponectin.microgm.ml...5000.X. Leptin..ng.ml.       HOMA2..B
##  Min.   : 0.00274                 Min.   :  1.558   Min.   :  1.70
##  1st Qu.: 1.71082                 1st Qu.:  7.412   1st Qu.: 31.40
##  Median : 3.40912                 Median : 16.770   Median : 53.00
##  Mean   : 3.89672                 Mean   : 28.011   Mean   : 57.36
##  3rd Qu.: 5.67910                 3rd Qu.: 36.532   3rd Qu.: 79.60
##  Max.   :12.32316                 Max.   :207.100   Max.   :168.30
##  NA's   :274                      NA's   :219
##     HOMA2..S        HOMA2.IR      Body.Fat....from.Age..BMI BMI_group
##  Min.   : 13.2   Min.   :0.1260   Min.   : 4.093            Lean :405
##  1st Qu.:105.0   1st Qu.:0.3186   1st Qu.:21.421            Obese:245
##  Median :187.6   Median :0.5331   Median :27.890
##  Mean   :228.0   Mean   :0.7808   Mean   :28.044
##  3rd Qu.:313.9   3rd Qu.:0.9526   3rd Qu.:33.375
##  Max.   :793.6   Max.   :7.5758   Max.   :71.180
##
##       FBS              Ins            HOMA_IR            HOMA_B
##  Min.   : 65.97   Min.   : 1.021   Min.   : 0.1804   Min.   :  1.509
##  1st Qu.: 82.41   1st Qu.: 2.345   1st Qu.: 0.5664   1st Qu.: 22.407
##  Median : 94.05   Median : 3.836   Median : 0.9713   Median : 49.166
##  Mean   :115.94   Mean   : 5.576   Mean   : 1.7164   Mean   : 62.917
##  3rd Qu.:128.87   3rd Qu.: 7.099   3rd Qu.: 1.9248   3rd Qu.: 89.232
##  Max.   :379.11   Max.   :36.245   Max.   :18.6231   Max.   :233.724
##
```

Representamos histogramas de frecuencias de variables que nos puede interesar conocer cómo se distribuyen. La función hist.dataframe del paquete Hmisc hace histogramas con todas las variables del dataframe

```
hist.data.frame(datosdf)
```

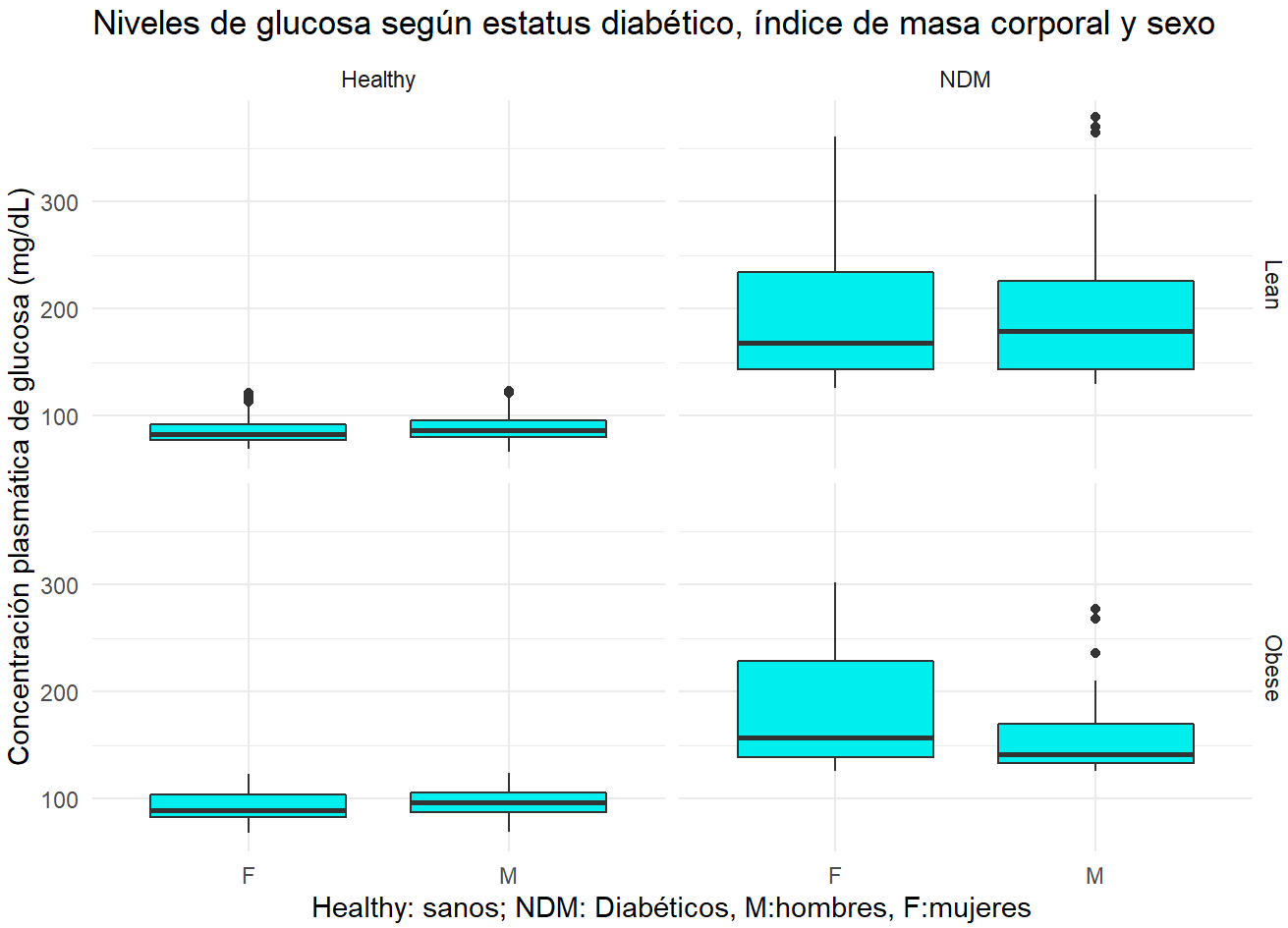



```
#A partir de la anterior, seleccionamos algunas variables que puedan
interesar más:
par(mfrow=c(3,3))
hist(FBS, main="Glucosa plasmática", breaks=50, col="cadetblue1")
hist(TG, main="Triglicéridos", breaks=50, col="cadetblue2")
hist(TC, main="Colesterol total", breaks=50, col="cadetblue3")
hist(Ins, main = "Insulina", breaks=50, col="olivedrab1")
hist(Adiponectin.microgm.ml...5000.X., main = "Adiponectina", breaks=50,
col="olivedrab2")
hist(Leptin..ng.ml., main = "Leptina", breaks=50, col="olivedrab3")
```



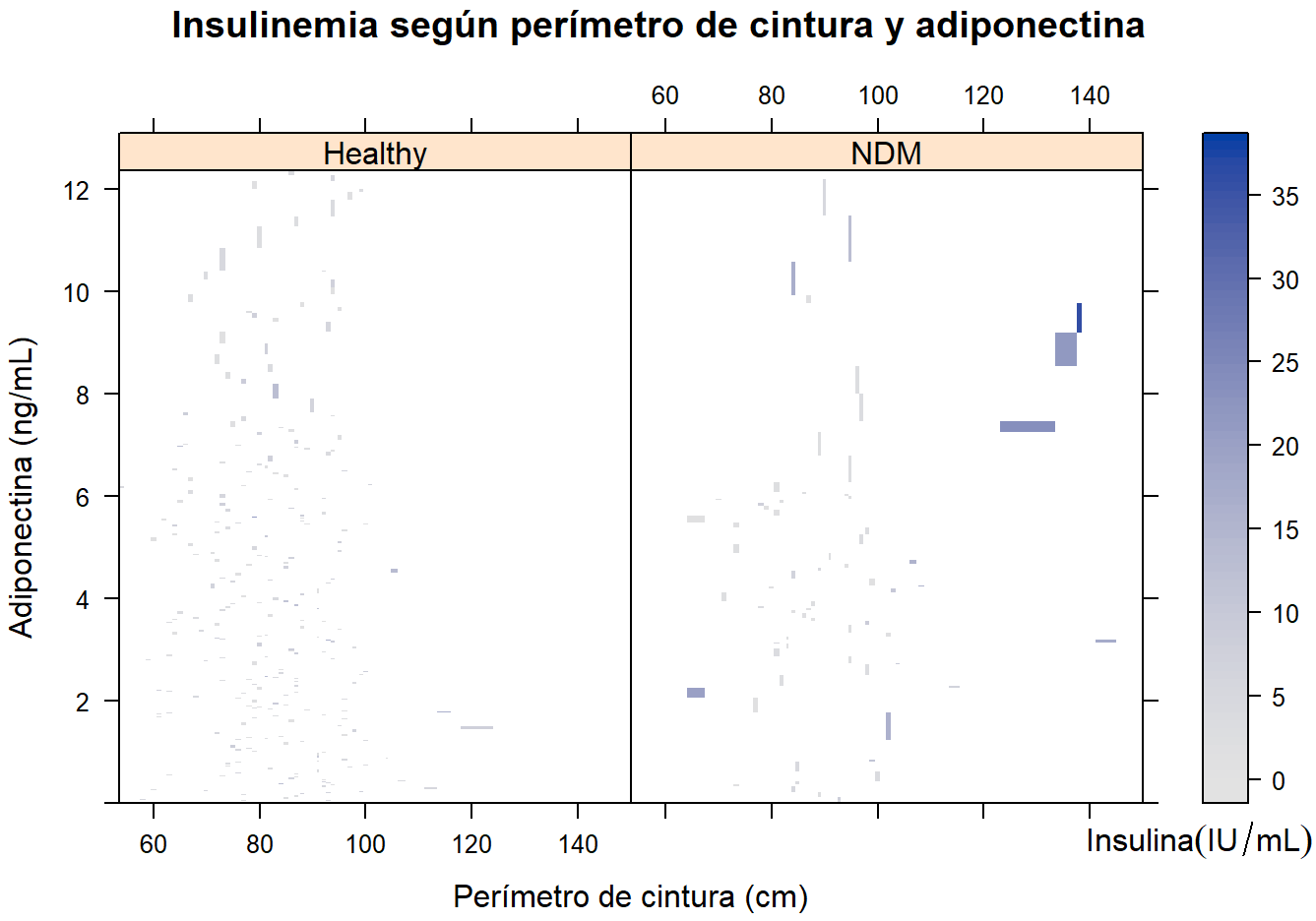
Representamos diagramas de cajas que ayuden a visualizar gráficamente las tablas de medias que hemos construido en los apartados anteriores. Usamos ggplot2 con un facet_grid, que es muy útil para dividir por variables los gráficos de manera sencilla:

```
#Boxplots glucosa plasmática según estatus diabético, índice de masa corporal
y sexo
box_FBS<-ggplot(datosdf, aes(x=SEX, y=FBS))+
  geom_boxplot(fill="cyan2")+
  facet_grid(BMI_group~DM_status)+
  labs(title ="Niveles de glucosa según estatus diabético, índice de masa
corporal y sexo", x = "Healthy: sanos; NDM: Diabéticos, M:hombres,
F:mujeres", y = "Concentración plasmática de glucosa (mg/dL)")+
  theme_minimal()
box_FBS
```

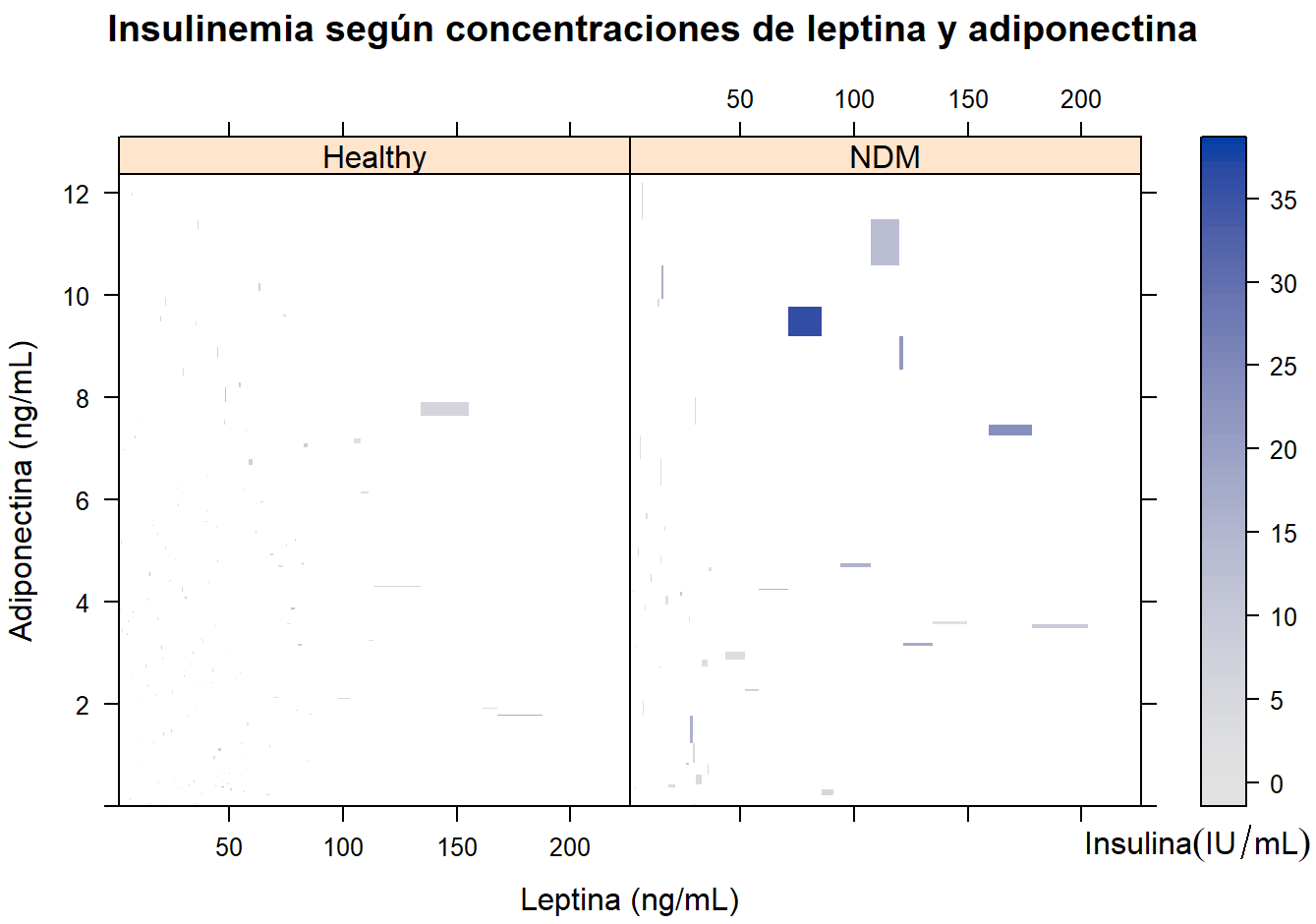


Representamos gráficos con 3 variables usando levelplot, una función incluída en el paquete lattice. Construye gráficos de niveles en un diagrama 2D en el que la tercera varaibles se representa mediante una escala de colores. Para poder añadir una leyenda a la escala de colores es necesario usar una función adicional, trellis.focus. Usamos estas representaciones para visualizar niveles de insulina en según el estado diabético y las distintas variables asociadas a adiposidad (WC, BMI, leptina y adiponectina):

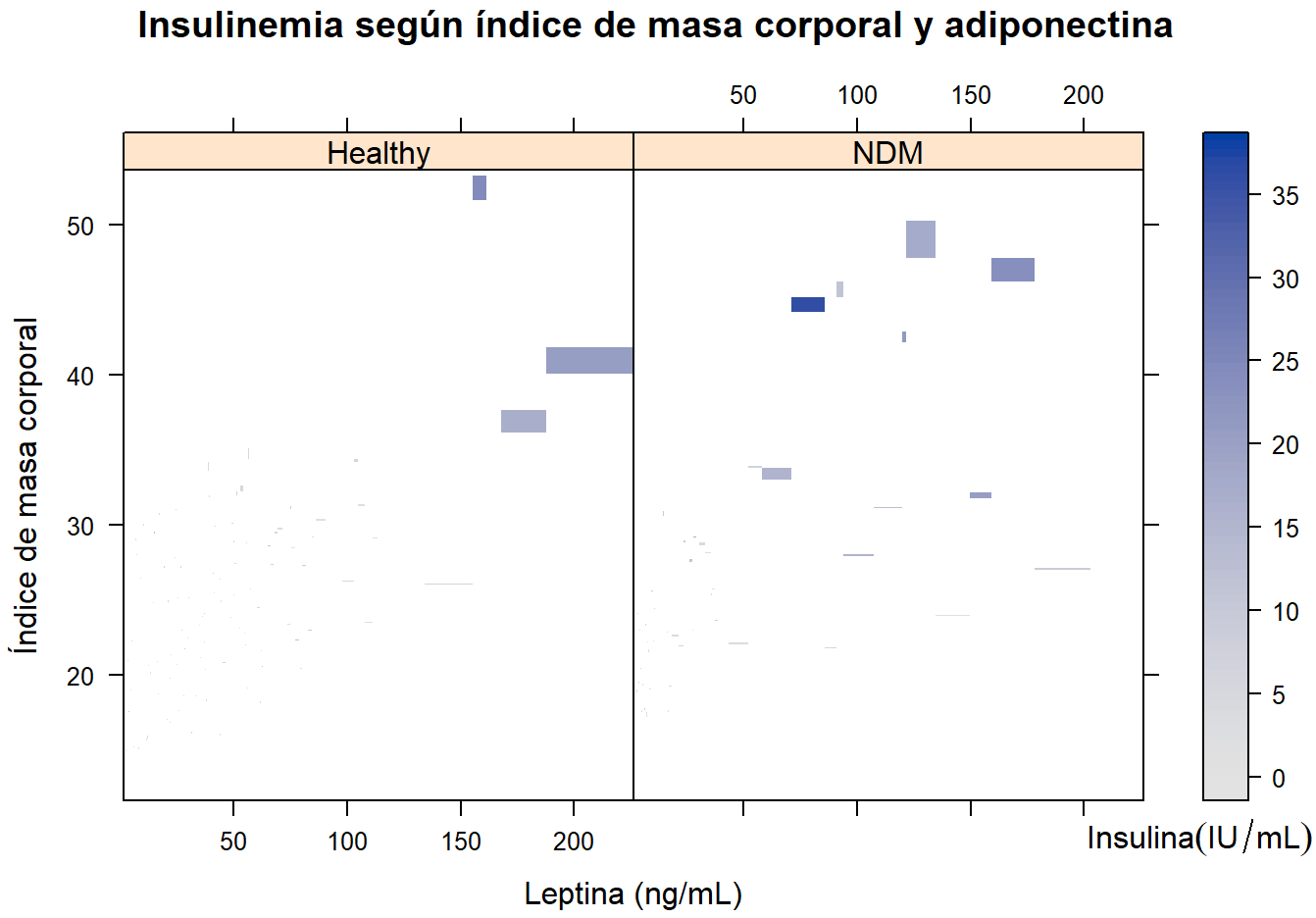
```
#Levelplot multivariable: Insulinemia según distintas combinaciones de
parámetros de adiposidad, para diabéticos y no diabéticos).
levelplot(Ins~WC*Adiponectin.microgm.ml...5000.X.|DM_status, data=datosdf,
cuts = 80, col.regions = sequential_hcl(100, rev = T),
form="fit", xlab = "Perímetro de cintura (cm)", ylab = "Adiponectina
(ng/mL)", main=" Insulinemia según perímetro de cintura y adiponectina")
trellis.focus("legend", side="right", clipp.off=TRUE, highlight=FALSE)
grid::grid.text(expression(Insulina (IU/mL)), 0.2, 0, hjust=0.6, vjust=1.5)
```




```
levelplot(Ins~Leptin..ng.ml.*Adiponectin.microgm.ml...5000.X.|DM_status,
data=datosdf, cuts = 80, col.regions = sequential_hcl(100, rev = T),
form="fit", xlab = "Leptina (ng/mL)", ylab = "Adiponectina (ng/mL)",
main="Insulinemia según concentraciones de leptina y adiponectina")
trellis.focus("legend", side="right", clipp.off=TRUE, highlight=FALSE)
grid::grid.text(expression(Insulina (IU/mL)), 0.2, 0, hjust=0.6, vjust=1.5)
```

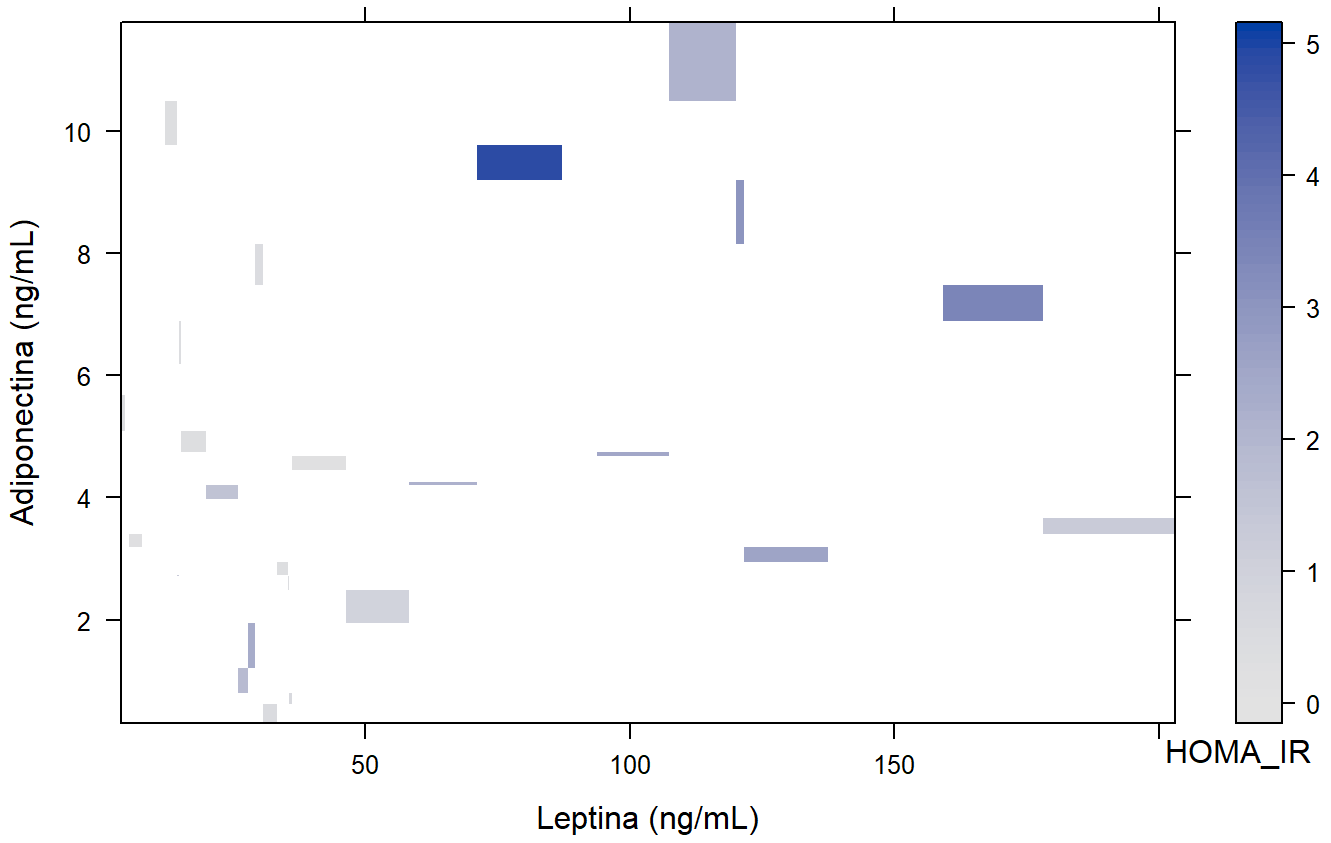



```
levelplot(Ins~Leptin..ng.ml.*BMI|DM_status, data=datosdf, cuts = 80,
col.regions = sequential_hcl(100, rev = T),
form="fit", xlab = "Leptina (ng/mL)", ylab = "Índice de masa corporal",
main="Insulinemia según índice de masa corporal y adiponectina")
trellis.focus("legend", side="right", clipp.off=TRUE, highlight=FALSE)
grid::grid.text(expression(Insulina (IU/mL)), 0.2, 0, hjust=0.6, vjust=1.5)
```



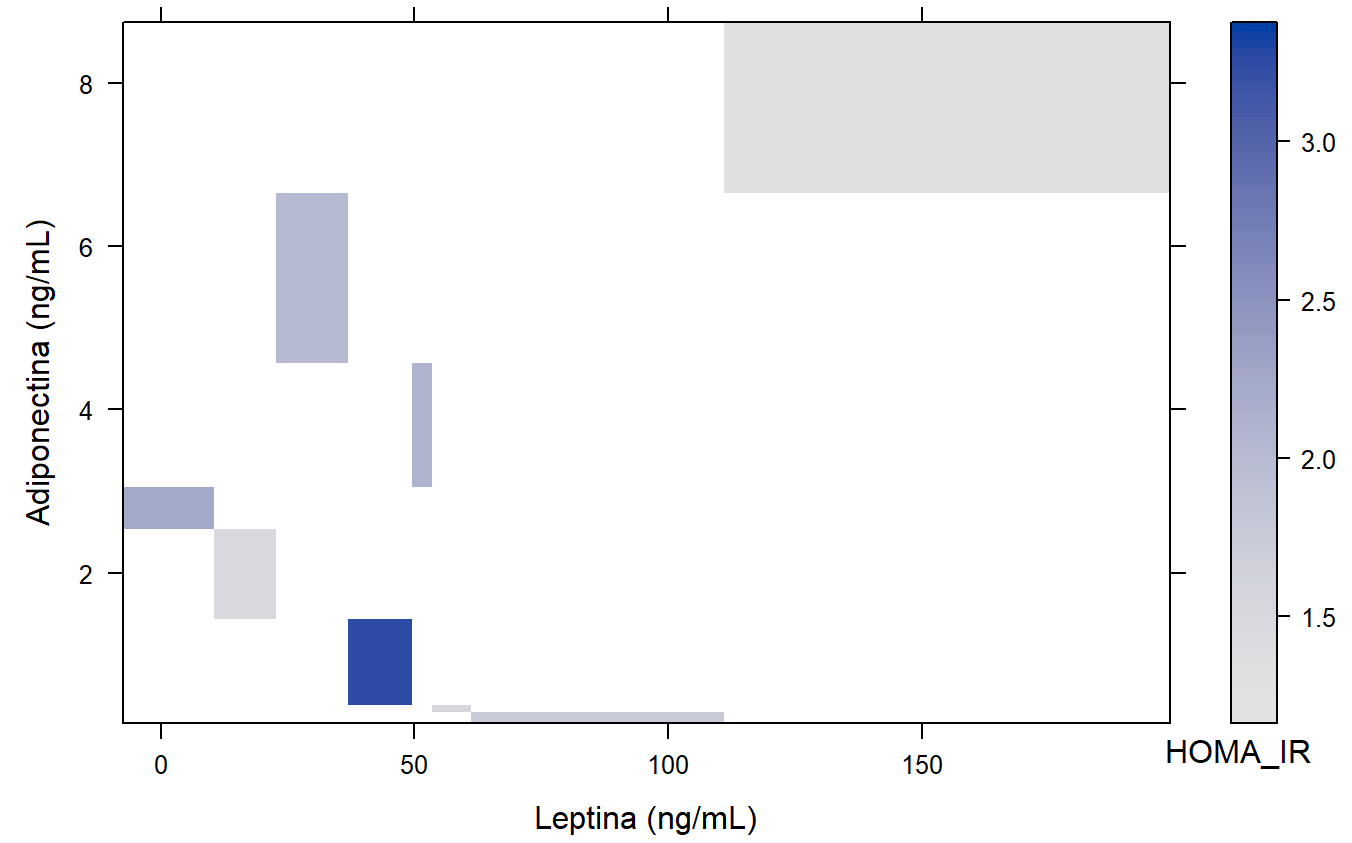

```
#Podemos afinar más si usamos un subset como el de los datos bioquímicos de
los individuos diabéticos y obesos. Representamos en este caso la resistencia
a insulina (índice HOMA.IR) según niveles de leptina y adiponectina, pero
dentro del grupo de los obesos con diabetes:

levelplot(HOMA2.IR~Leptin..ng.ml.*Adiponectin.microgm.ml...5000.X.,
data=bioq_DM_Ob, cuts = 80, col.regions = sequential_hcl(100, rev = T),
form="fit", xlab = "Leptina (ng/mL)", ylab = "Adiponectina (ng/mL)",
main="HOMA.IR según leptina y adiponectina en diabéticos obesos")
trellis.focus("legend", side="right", clipp.off=TRUE, highlight=FALSE)
grid::grid.text(expression(HOMA_IR), 0.2, 0, hjust=0.6, vjust=1.5)
```




```
#Y compararlo con los obesos no diabéticos pero con síndrome metabólico:

levelplot(HOMA_IR~Leptin..ng.ml.*Adiponectin.microgm.ml...5000.X.,
data=bioq_nDM_MS_Ob, cuts = 80, col.regions = sequential_hcl(100, rev = T),
form="fit", xlab = "Leptina (ng/mL)", ylab = "Adiponectina (ng/mL)",
main="HOMA.IR según leptina y adiponectina en no diabético obesos con s.
metabólico")
trellis.focus("legend", side="right", clipp.off=TRUE, highlight=FALSE)
grid::grid.text(expression(HOMA_IR), 0.2, 0, hjust=0.6, vjust=1.5)
trellis.unfocus()
```



Estructura de los datos

```
str(datosdf)
## 'data.frame':    650 obs. of  23 variables:
##  $ þÿ                              : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ AGE                             : int  47 57 46 51 42 38 55 43 45 36
...
##  $ SEX                             : Factor w/ 2 levels "F","M": 2 1 2 2 2
1 1 1 2 1 ...
##  $ BMI                             : num  24.3 23.7 19.9 25.8 21.8 ...
##  $ WC                              : num  101 81 79 95 89 115 100 NA 86
129 ...
##  $ FBS..mmol.l.                    : num  4.29 4.76 5.29 4.84 5.12 ...
##  $ TG                              : num  199.1 69.8 115.6 219.1 267.4 ...
##  $ TC                              : num  146 184 215 208 213 ...
##  $ SBP                             : int  NA NA NA NA NA NA NA NA NA NA
...
##  $ DBP                             : int  NA NA NA NA NA NA NA NA NA NA
...
##  $ DM_status                       : Factor w/ 2 levels "Healthy","NDM": 1
1 1 1 1 1 1 2 1 1 ...
```

```
##  $ F.Ins.pmol.L.                    : num  33.3 29.6 22 33.6 62.8 ...
##  $ Adiponectin.microgm.ml...5000.X.: num  NA NA 1.65 NA NA ...
##  $ Leptin..ng.ml.                   : num  NA NA NA NA NA ...
##  $ HOMA2..B                         : num  94.3 70.7 46.8 74.2 101.3 ...
##  $ HOMA2..S                         : num  166.6 181.5 237.3 159.6 84.8 ...
##  $ HOMA2.IR                         : num  0.6 0.551 0.421 0.627 1.179 ...
##  $ Body.Fat....from.Age..BMI        : num  19.1 31.5 12.4 21.9 14.7 ...
##  $ BMI_group                        : Factor w/ 2 levels "Lean","Obese": 1
1 1 2 1 2 2 2 1 2 ...
##  $ FBS                              : num  77.2 85.7 95.2 87.2 92.1 ...
##  $ Ins                              : num  4.79 4.27 3.17 4.84 9.04 ...
##  $ HOMA_IR                          : num  0.914 0.903 0.745 1.041 2.057
...
##  $ HOMA_B                           : num  121.3 67.7 35.5 72 111.8 ...
```

Valores NA.

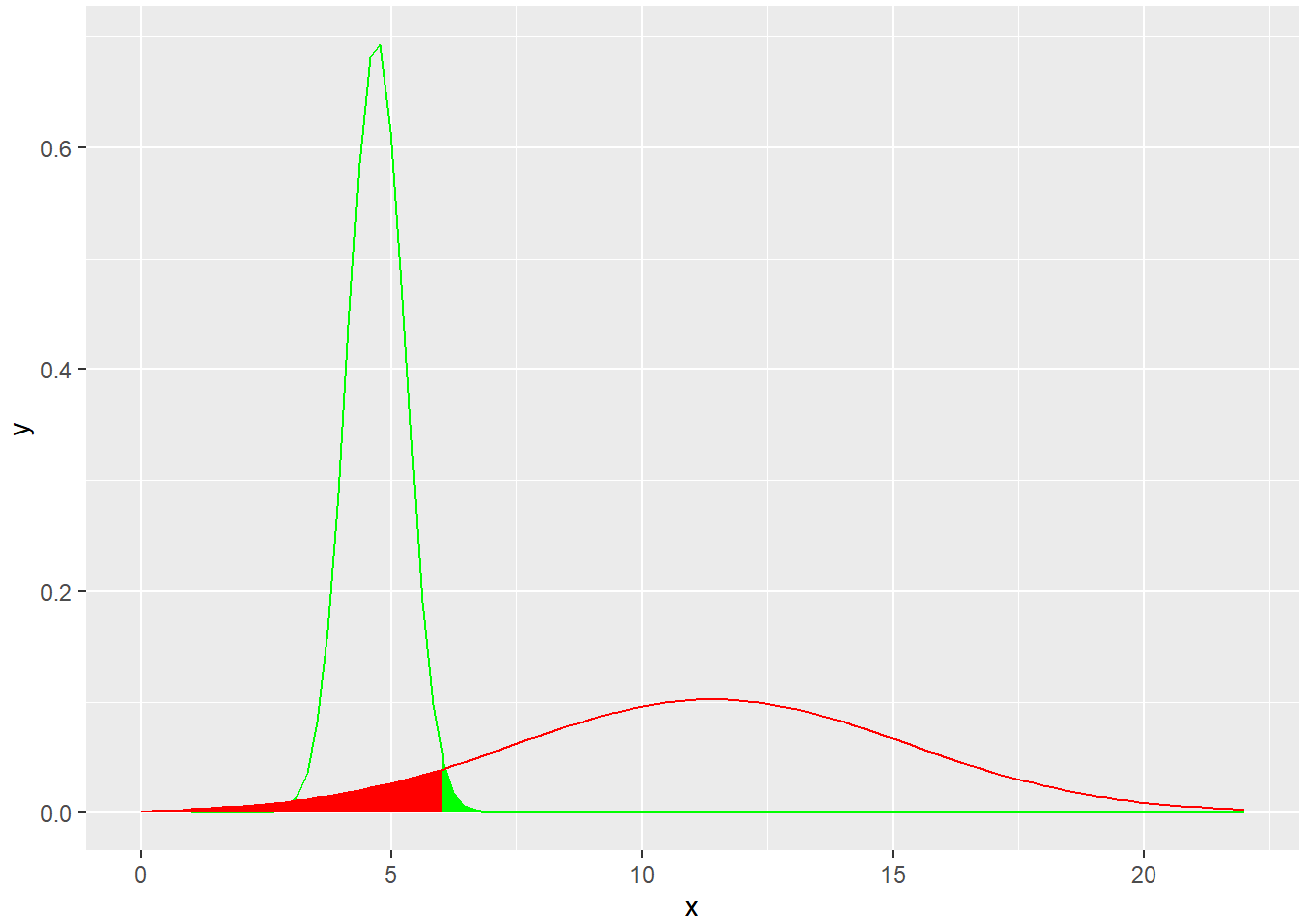
```
table(is.na(datosdf$AGE))
##
## FALSE  TRUE
##   649     1
datosdf2 =na.omit(datosdf)
table(is.na(datosdf2))
##
## FALSE
##  5819
```

(5) PROBABILIDAD (1p)

Distribución normal.

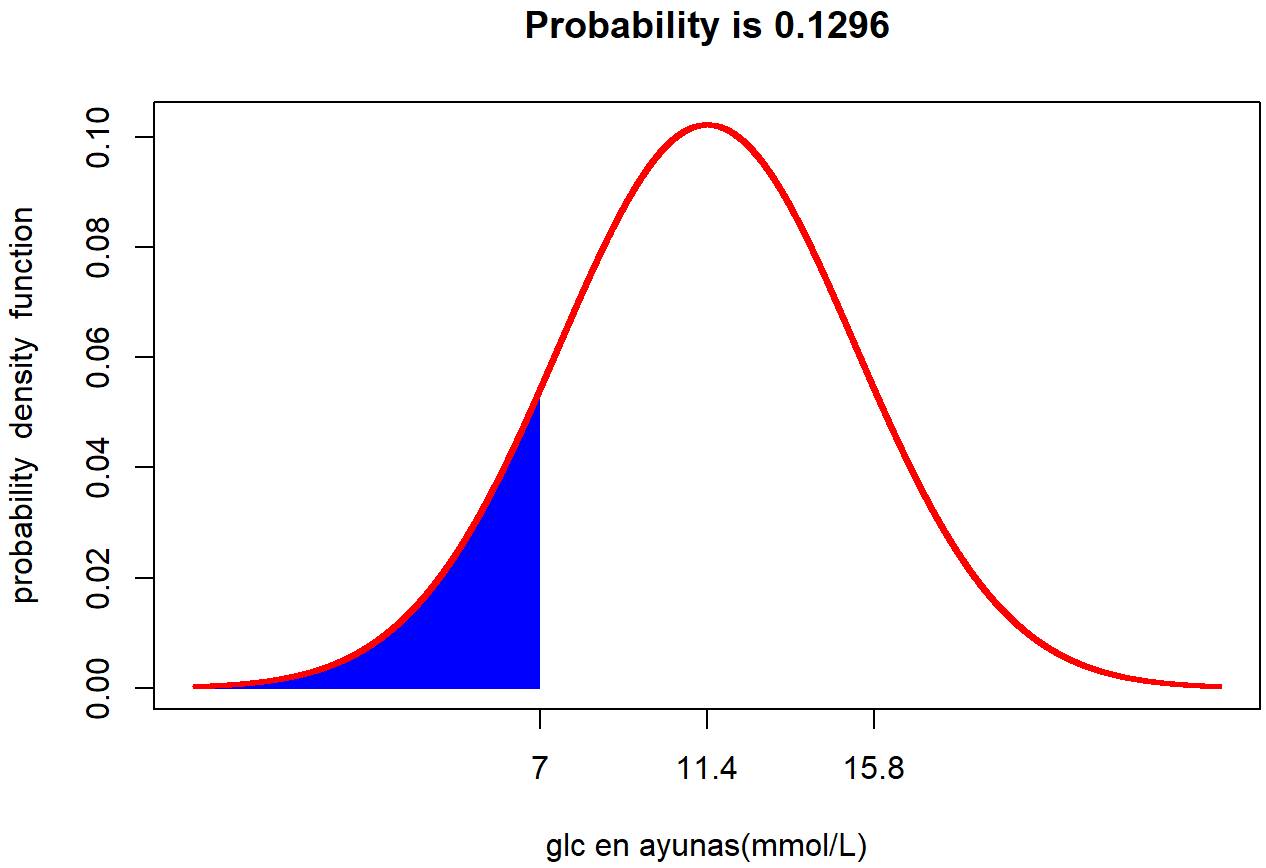
Tomamos como variable aleatoria la concentración de azúcar en sangre en ayunas, para dos grupos: enfermos o sanos. Suponemos que estos valores siguen una distribución normal en ambos grupos: SANOS: N(4.7, 0.57) DM2: N(11.4,3.9)

```
FBSH<- subset(datosdf2$FBS..mmol.l., datosdf2$DM_status=="Healthy")
FBSDM <- subset(datosdf2$FBS..mmol.l., datosdf2$DM_status=="NDM")
mean(FBSH); sd(FBSH); mean(FBSDM); sd(FBSDM)
## [1] 4.713599
## [1] 0.5677863
## [1] 11.42383
## [1] 3.918924
ggplot(data.frame(x = c(1, 22)), aes(x)) +
  stat_function(fun = dnorm, args = list(mean = 4.7, sd = 0.57), col='green')
+
  stat_function(fun = dnorm, args = list(mean =4.7, sd =.57), xlim = c(6,7),
                geom = "area", fill = "green")+
  stat_function(fun = dnorm, args = list(mean = 11.4, sd = 3.9), col='red') +
  stat_function(fun = dnorm, args = list(mean= 11.4, sd =3.9), xlim = c(0,6),
                geom = "area", fill = "red")
```



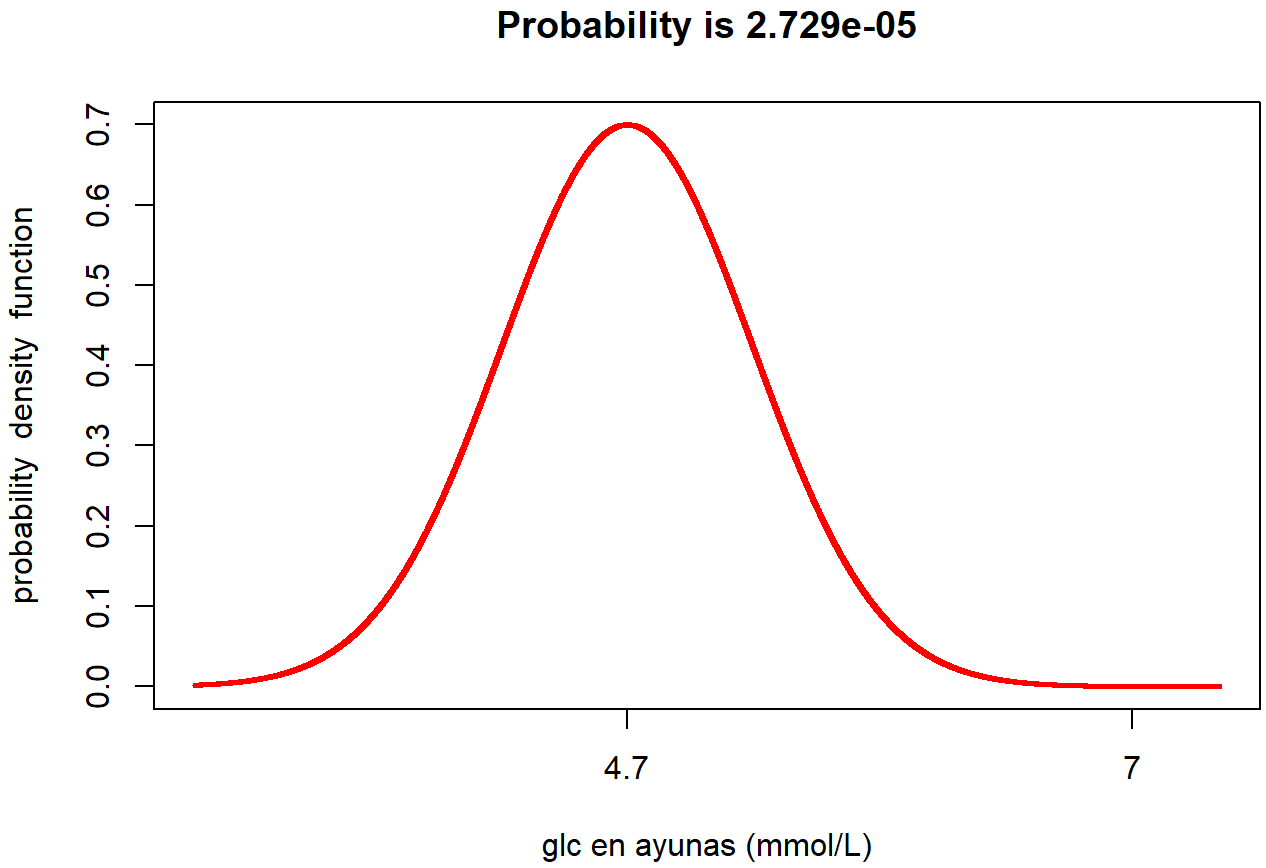
P1. Se considera que los individuos con valores de azúcar en sangre por encima de 7 serían clasificados como DM2. ¿qué porcentaje de la población se consideraría incorrectamente sana?

```
P1<- pnorm(7,mean=11.4,sd=3.9, lower.tail = T)
cat("la probabilidad de ser clasificado incorrectamente sano es:", P1)
## la probabilidad de ser clasificado incorrectamente sano es: 0.1296166
shadeDist(7,"dnorm",11.4, 3.9,lower.tail = T, col=c("red", "blue"), xlab="glc
en ayunas(mmol/L)")
```



P2 ¿qué porcentaje se clasificaría como enfermo pero está realmente sano?

```
P2<- pnorm(7, mean=4.7, sd=0.57, lower.tail = F)
cat("la probabilidad de ser clasificado erróneamente como DM2 es:", P2)
## la probabilidad de ser clasificado erróneamente como DM2 es: 2.729095e-05
shadeDist(7, "dnorm", 4.7, 0.57, lower.tail = F, col=c("red","blue"),
xlab="glc en ayunas (mmol/L)")
```



P3. ¿Cuál es la probabilidad en el grupo de individuos con DM2 que se obtengan concentraciones entre 10 y 13?

```
pnorm(c(13,10), mean = 11.4, sd=3.9)
## [1] 0.6591911 0.3598071
cat("la probabilidad de obtener concentraciones entre 10 y 13 en DM2 es: ",
0.6592-0.3598)
## la probabilidad de obtener concentraciones entre 10 y 13 en DM2 es:
0.2994
```

P4. Determinar la concentración mínima del 30% de los individuos sanos con más concentración.

Equivale a calcular el quantil 25:

```
qnorm(0.3, mean=4.7, sd=0.57)
## [1] 4.401092
```

es decir, P[X < 4.4] = 0.30

Binomial.

P5. Si la probabilidad de tener DM2 es de un 8%, ¿que probabilidad hay de que si vienen 16 voluntarios a hacerse la analítica sean todos diabéticos? Se trataria entonces de una distribución

binomial: n = 16 y p = 0.08. (X → B (16, 0.08))

```
td<- dbinom(16,16,0.08)
cat("P[X=16]=",td)
## P[X=16]= 2.81475e-18
```

Es prácticamente 0.

P6. ¿y de que más de la mitad lo sean?

```
md<- pbinom(8,16,0.08)
cat("P[Sean diabéticos tipo 2 más de la mitad] = P[X > 8] = 1 - P[X =< 8] = 1
- F(8)=", 1-md)
## P[Sean diabéticos tipo 2 más de la mitad] = P[X > 8] = 1 - P[X =< 8] = 1 -
F(8)= 9.112486e-07
```

También es muy poco probable.

P7. Y que ninguno sea diabético tipo 2?

```
nd<- dbinom(0,16,0.08)
cat("P[ninguno sea diabético 2] = P[X = 0]=", nd)
## P[ninguno sea diabético 2] = P[X = 0]= 0.2633936
```

P8. ¿Y la probabilidad de que, al menos, 2 sean diabéticos tipo2?

```
dd<- 1-pbinom(1,16,0.08)
cat("P[por lo menos dos sean diabéticos] = P[X >= 2] = 1 - P[X < 2] = 1 -
F(1)=", dd)
## P[por lo menos dos sean diabéticos] = P[X >= 2] = 1 - P[X < 2] = 1 - F(1)=
0.3701457
```

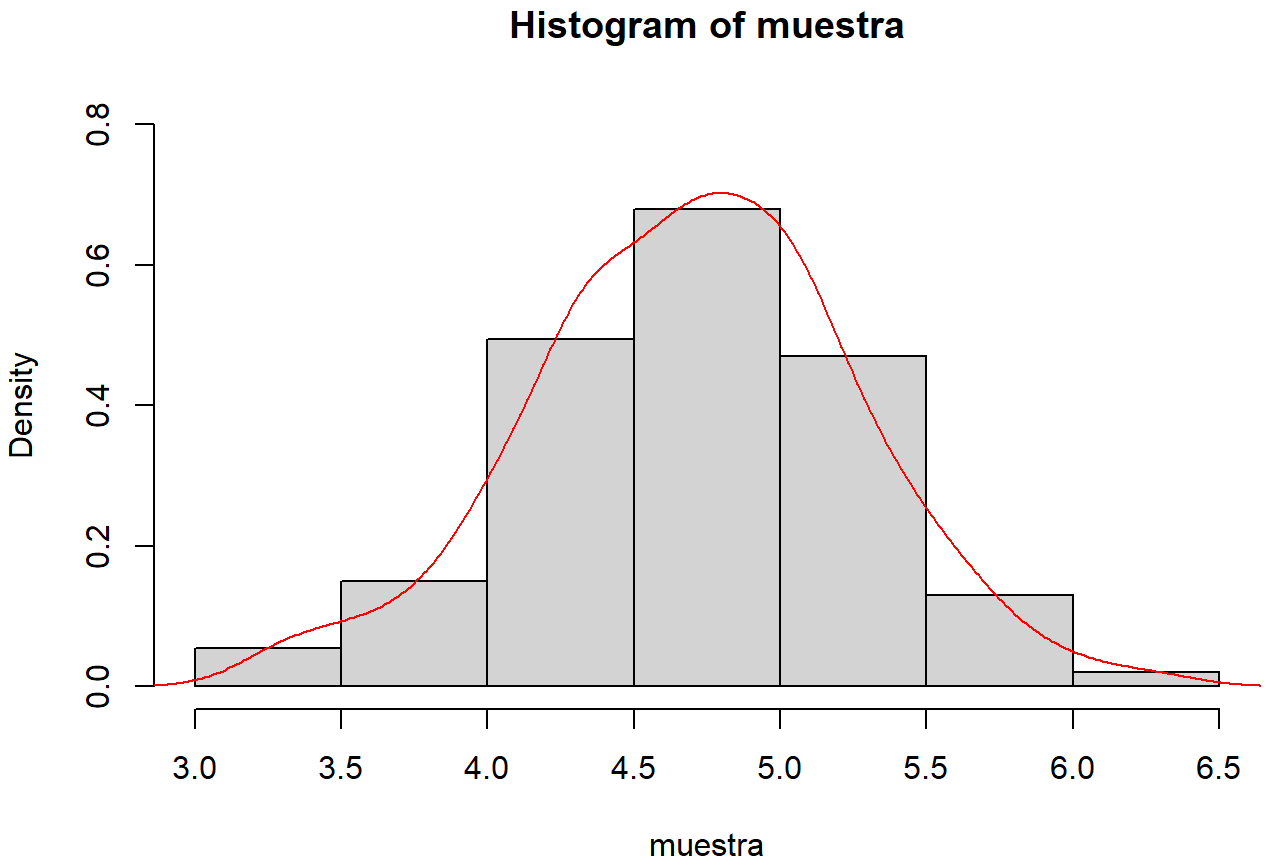
P9. Generar una muestra de tamaño 40 de la población con DM2.

P10. Estimar la media de la muestra con su intervalo de confianza al 95%.

```
set.seed(2027)
muestra<- rnorm(400, mean=4.7, sd=0.57)
media40<- mean(muestra)
sd40<- sd(muestra)
se40<- sd40/sqrt(length(muestra))
li40<- media40-qt(.975, length(muestra)-1)*se40
ls40<- media40+qt(.975, length(muestra)-1)*se40
cat("el intervalo de confianza es:",li40, ls40)
## el intervalo de confianza es: 4.647513 4.757533
```

También se puede calcular con un t-test:

```
t.test(muestra, conf.level = 0.95)
##
##  One Sample t-test
##
## data:  muestra
## t = 168.06, df = 399, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  4.647513 4.757533
## sample estimates:
## mean of x
##  4.702523
hist(muestra,ylim=c(0,0.8), freq = F)
lines(density(muestra), col="red")
```



(6) REGRESIÓN LINEAL (1p)

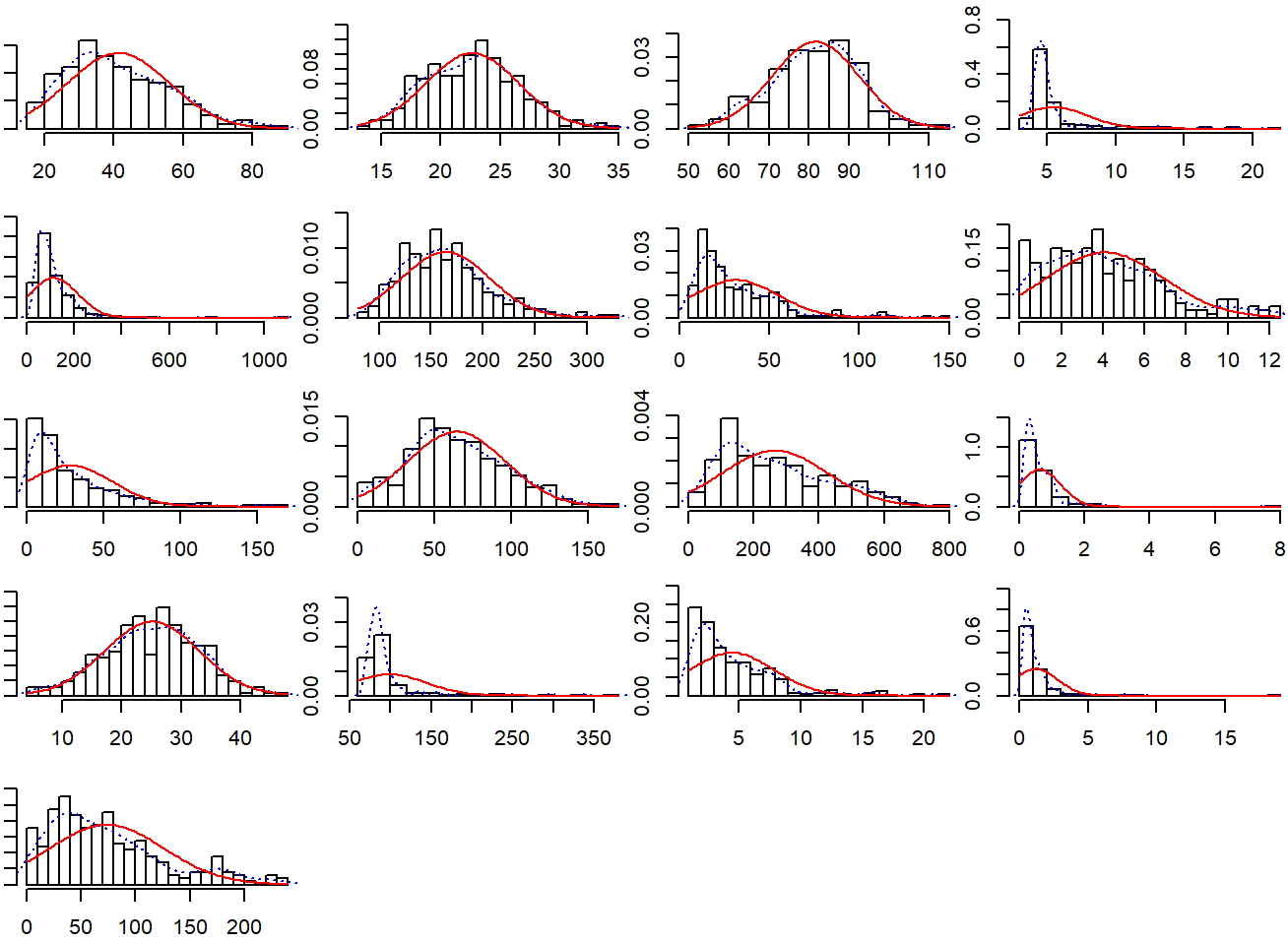
HOMA IR= (insul.en sangre en ayuno x glucosa en sangre en ayuno)/22.5

HOMA B= 20X insul. en sangre en ayuno x /(gluc. en ayuno-3.5)

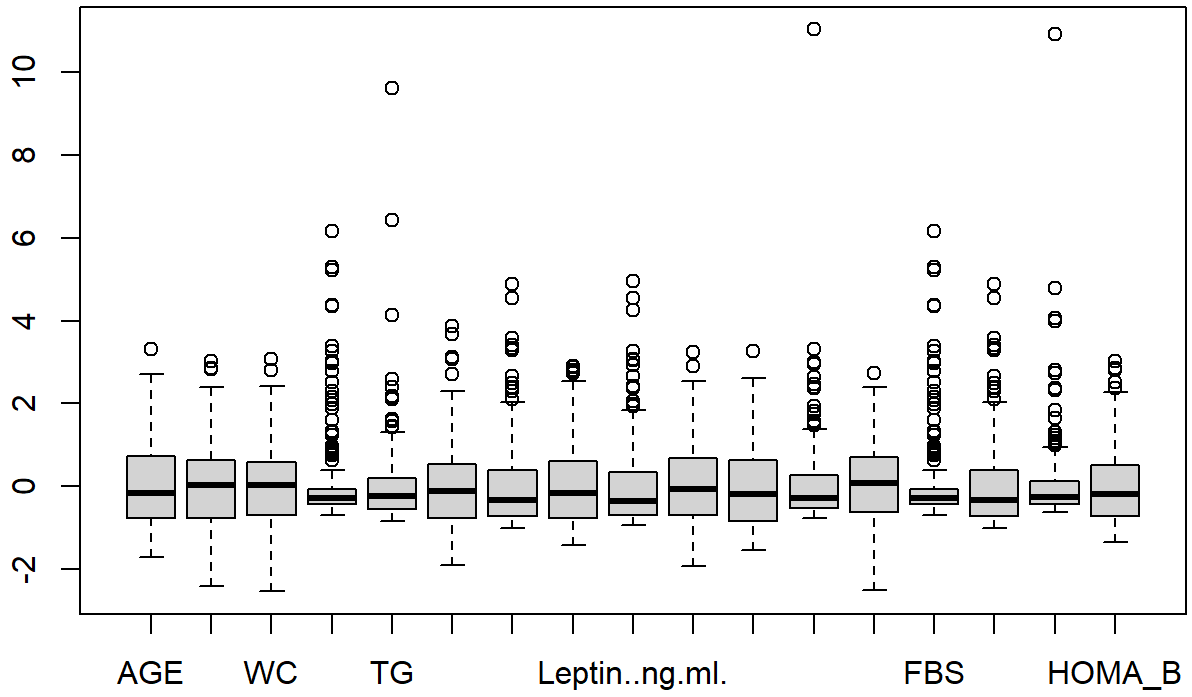
Los no obesos son individuos con IMC <25(La OMS define a un adulto que tenga un BMI entre 25 y 29,9 como exceso de peso - adulto que tenga un BMI de 30 o más alto se considera obeso - un BMI abajo de 18,5 se considere peso insuficiente, y entre 18,5 a 24,9 al peso sano).

Normalidad de las variables:

```
multi.hist(x = datosdf2[,c(2,4,5,6,7,8,12,13,14,15,16,17,18,20,21,22,23)],
dcol = c("blue","red"), dlty = c("dotted", "solid"),
          main = "")
```




```
mvn(data=datosdf2[,c(2,4,5,6,7,8,12,13,14,15,16,17,18,20,21,22,23)],
mvnTest="royston", univariatePlot="box")
```



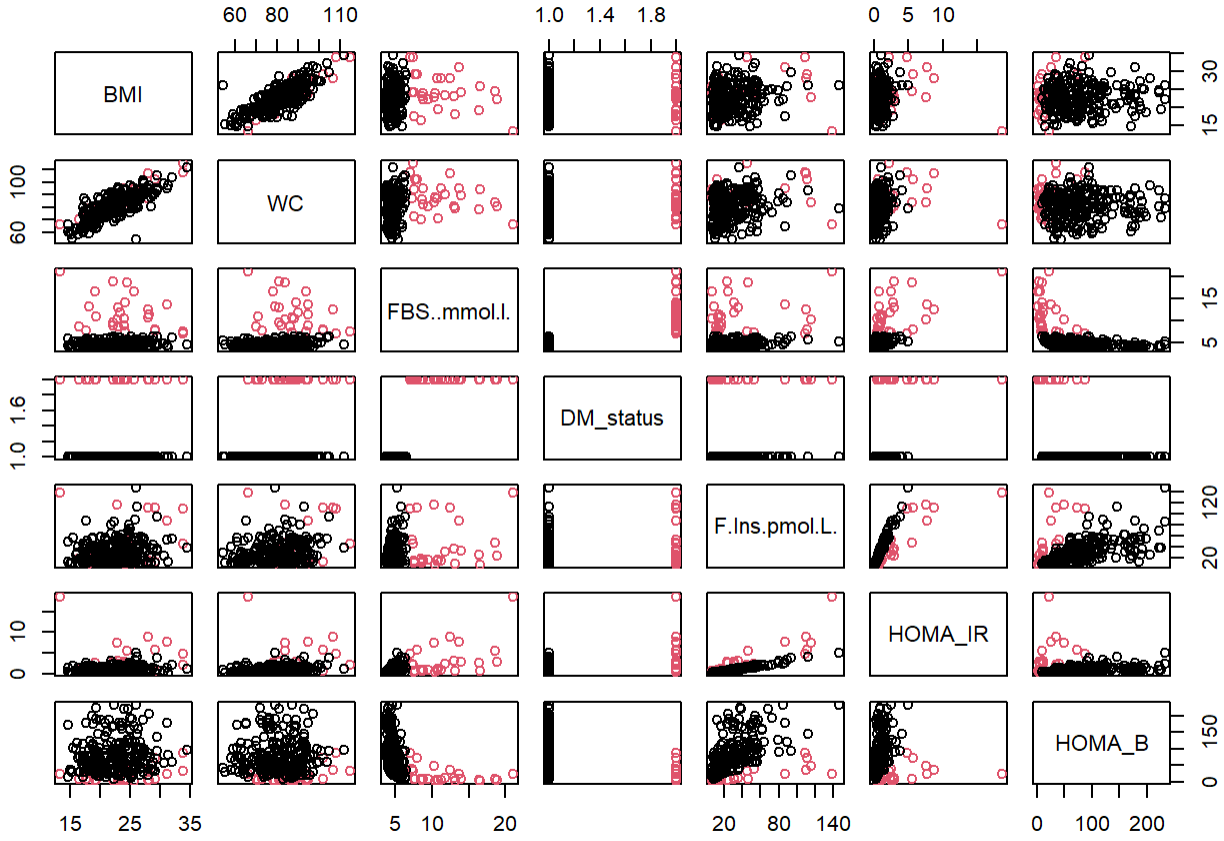

```
## $multivariateNormality
##      Test       H       p value MVN
## 1 Royston 554.683 1.418198e-112  NO
##
## $univariateNormality
##            Test                          Variable Statistic   p value
Normality
## 1  Shapiro-Wilk                AGE                    0.9656  <0.001      NO
## 2  Shapiro-Wilk                BMI                    0.9918  0.1732
YES
## 3  Shapiro-Wilk                 WC                    0.9920  0.1884
YES
## 4  Shapiro-Wilk           FBS..mmol.l.                0.5317  <0.001      NO
## 5  Shapiro-Wilk                 TG                    0.5892  <0.001      NO
## 6  Shapiro-Wilk                 TC                    0.9544  <0.001      NO
## 7  Shapiro-Wilk          F.Ins.pmol.L.                0.8052  <0.001      NO
## 8  Shapiro-Wilk Adiponectin.microgm.ml...5000.X.      0.9442  <0.001      NO
## 9  Shapiro-Wilk          Leptin..ng.ml.               0.7914  <0.001      NO
## 10 Shapiro-Wilk             HOMA2..B                  0.9847  0.0082      NO
## 11 Shapiro-Wilk             HOMA2..S                  0.9338  <0.001      NO
## 12 Shapiro-Wilk             HOMA2.IR                  0.5683  <0.001      NO
```

```
## 13 Shapiro-Wilk    Body.Fat....from.Age..BMI        0.9956  0.6873
YES
## 14 Shapiro-Wilk               FBS                   0.5317  <0.001      NO
## 15 Shapiro-Wilk               Ins                   0.8052  <0.001      NO
## 16 Shapiro-Wilk             HOMA_IR                 0.4712  <0.001      NO
## 17 Shapiro-Wilk              HOMA_B                 0.9126  <0.001      NO
##
## $Descriptives
##                                    n        Mean     Std.Dev      Median
## AGE                              253  41.3003953  14.6811576  39.0000000
## BMI                              253  22.6488131   3.9167164  22.7731484
## WC                               253  81.6166008  10.8638417  82.0000000
## FBS..mmol.l.                     253   5.4562331   2.5261406   4.7111753
## TG                               253 112.2032417 100.9147713  87.9544345
## TC                               253 164.0584785  42.2774065 159.1562189
## F.Ins.pmol.L.                    253  31.0304329  23.5908640  22.8819600
## Adiponectin.microgm.ml...5000.X. 253   4.0906575   2.8368091   3.6036301
## Leptin..ng.ml.                   253  27.7508751  27.9795394  17.5167024
## HOMA2..B                         253  64.3118577  32.0076048  62.0000000
## HOMA2..S                         253 263.9407115 162.0497424 234.1000000
## HOMA2.IR                         253   0.6138378   0.6306005   0.4271679
## Body.Fat....from.Age..BMI        253  25.0912828   8.0123349  25.6166667
## FBS                              253  98.2121960  45.4705305  84.8011561
## Ins                              253   4.4680249   3.3968127   3.2947387
## HOMA_IR                          253   1.1699371   1.5962017   0.7547540
## HOMA_B                           253  73.2624418  53.0448512  62.8068037
##                                          Min         Max         25th
## AGE                               16.00000000   90.000000  30.0000000
## BMI                               13.22314050   34.516765  19.6311176
## WC                                54.00000000  115.000000  74.0000000
## FBS..mmol.l.                       3.70158103   21.061825   4.3632103
## TG                                25.73750584 1084.235294  56.1990050
## TC                                83.50000000  327.864407 131.6425726
## F.Ins.pmol.L.                      7.09114132  146.420174  14.0353588
## Adiponectin.microgm.ml...5000.X.   0.01608613   12.323159   1.8888721
## Leptin..ng.ml.                     1.55794736  166.693800   8.0832492
## HOMA2..B                           2.20000000  168.300000  42.0000000
## HOMA2..S                          13.20000000  793.600000 128.7000000
## HOMA2.IR                           0.12600806    7.575758   0.2723312
## Body.Fat....from.Age..BMI          4.87837899   47.035148  19.9983790
## FBS                               66.62845850  379.112856  78.5377856
## Ins                                1.02104267   21.082818   2.0209300
## HOMA_IR                            0.18039267   18.623074   0.4584818
## HOMA_B                             1.63410897  233.723509  35.0146014
##                                         75th        Skew    Kurtosis
## AGE                               52.0000000  0.60814759 -0.07742782
## BMI                               25.1095690  0.24991358 -0.11254291
## WC                                88.0000000 -0.05431008 -0.03046716
## FBS..mmol.l.                       5.2932099  3.58199519 14.16669567
## TG                               132.3808339  5.14930182 39.70399193
## TC                               186.3310418  0.89835849  1.19218765
## F.Ins.pmol.L.                     40.3992206  1.95971848  4.85395319
## Adiponectin.microgm.ml...5000.X.   5.8114646  0.78684011  0.24340613
## Leptin..ng.ml.                    37.5741340  2.00859589  5.03300624
```

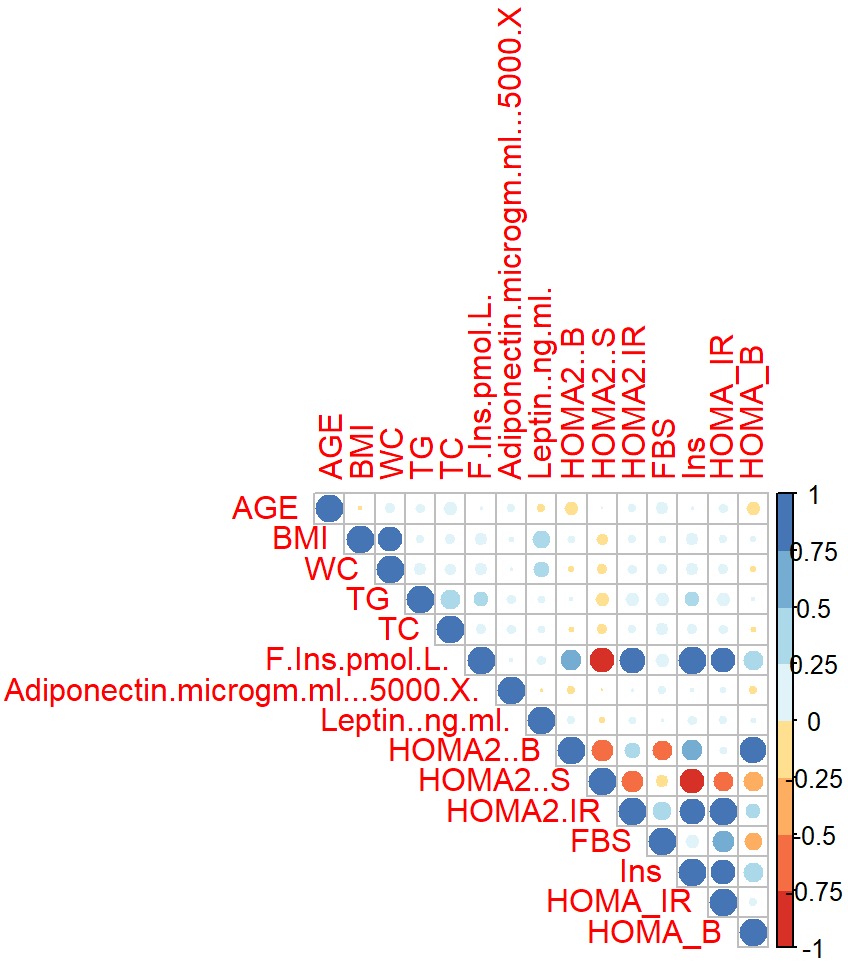
```
## HOMA2..B                          86.2000000  0.38728155 -0.09935371
## HOMA2..S                         367.2000000  0.72723033 -0.31394730
## HOMA2.IR                           0.7770008  5.99852639 57.72894323
## Body.Fat....from.Age..BMI         30.6440951 -0.11022298 -0.23855816
## FBS                               95.2777778  3.58199519 14.16669567
## Ins                                5.8170224  1.95971848  4.85395319
## HOMA_IR                            1.3680141  6.43794681 58.51189040
## HOMA_B                           100.3093973  1.00419093  0.41499867
# Con esta función obtenemos la información estadística básica y un Test
Shapiro-Wilk de normalidad de cada una de las variables.
```

Correlación entre las variables.

```
pairs(datosdf2[,c(4,5,6,11,12,22,23)], col=datosdf2$DM_status)
```




```
cordatos<- cor(datosdf2[,-c(1,3,6,9,10,11,18,19)])
corrplot(cordatos,type="upper", col = brewer.pal(n = 8, name = "RdYlBu"))
```

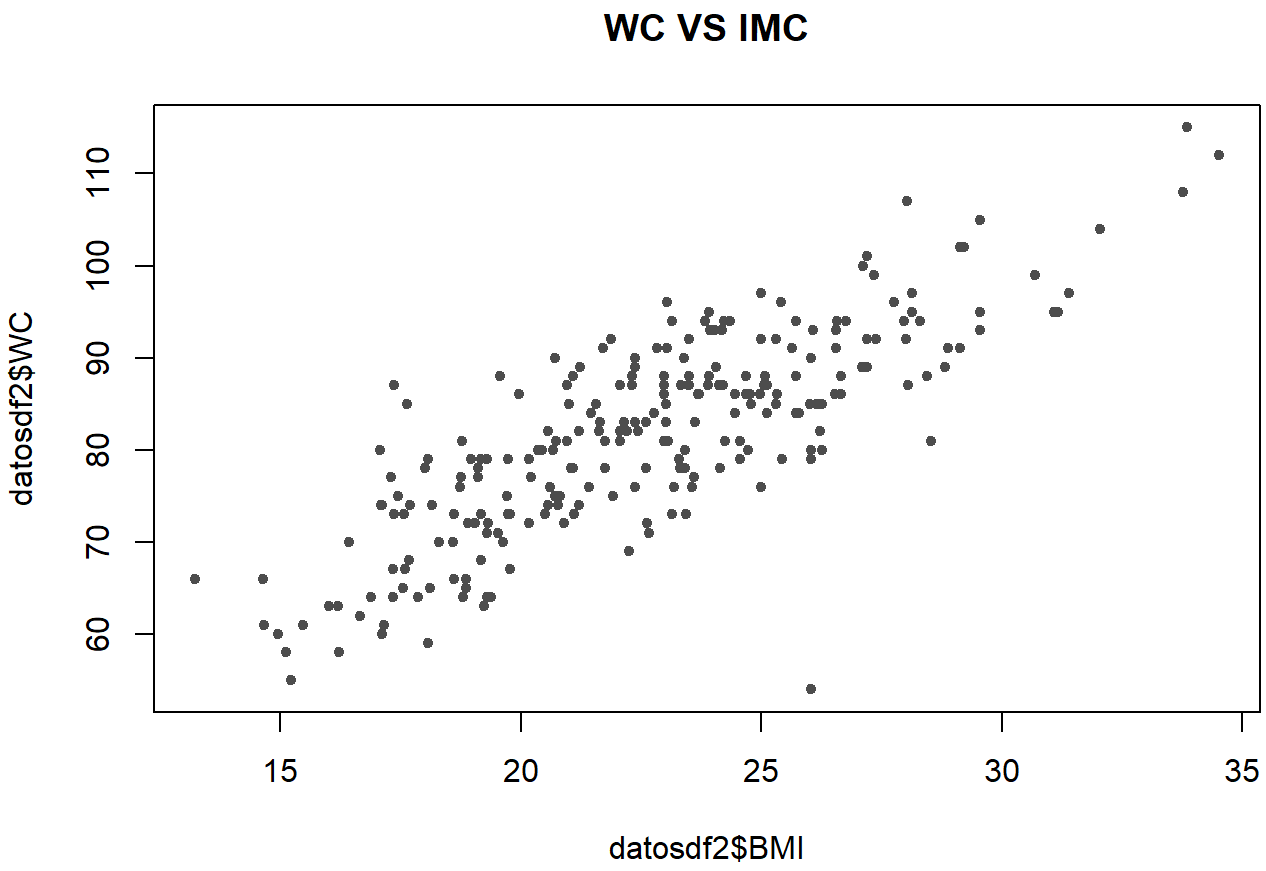


Observamos casos en los que la correlación es alta y por lo tanto aportan información redundante. Se excluirán algunos de estos predictores redundantes en el modelo: HOMA2B, HOMA2S, HOMA2.IR, Ins, FBS.

Modelo lineal simple.

Como se puede ver en el gráfico anterior (scatterplot), parece que existe una relación lineal entre el IMC y el perímetro de la cintura(WC):

```
MLS<-lm(data = datosdf2, formula =BMI~WC)
plot(x =datosdf2$BMI, y = datosdf2$WC, main = "WC VS IMC", pch = 20, col =
"grey30")
```




```
summary(MLS)
##
## Call:
## lm(formula = BMI ~ WC, data = datosdf2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.854 -1.534 -0.183  1.516 11.411
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.05973    1.10972  -0.955    0.341
## WC           0.29049    0.01348  21.552   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.324 on 251 degrees of freedom
## Multiple R-squared:  0.6492, Adjusted R-squared:  0.6478
## F-statistic: 464.5 on 1 and 251 DF,  p-value: < 2.2e-16
```

es decir, según el modelo, si mi cintura mide 65 cm, mi IMC sería:

```
0.29*65-1.06
```

```
## [1] 17.79
```

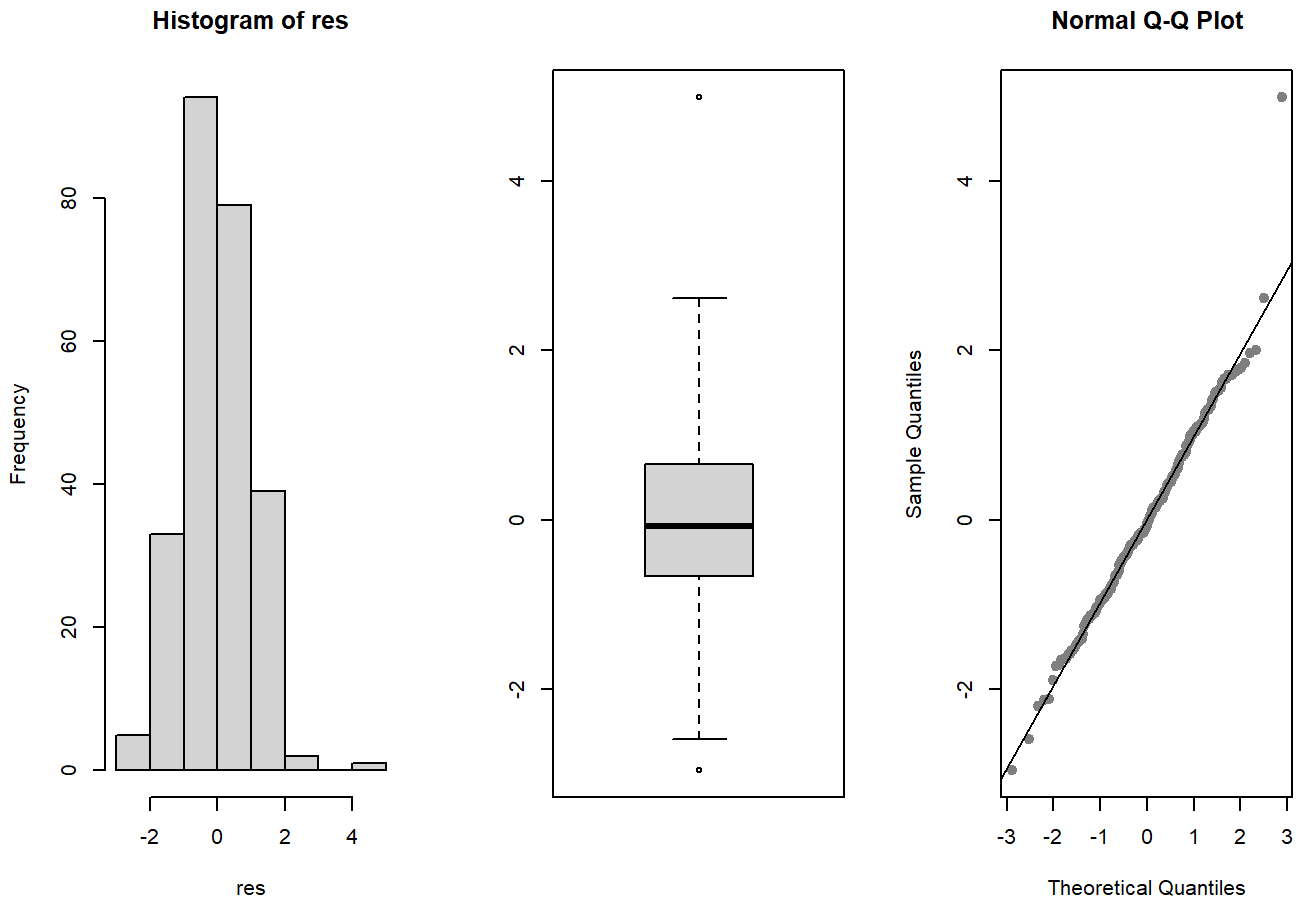
Test de Pearson.

```
cor.test(datosdf2$BMI, datosdf2$WC, method="pearson")
##
##  Pearson's product-moment correlation
##
## data:  datosdf2$BMI and datosdf2$WC
## t = 21.552, df = 251, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.757689 0.845080
## sample estimates:
##       cor
## 0.8057264
```

La correlación es alta.

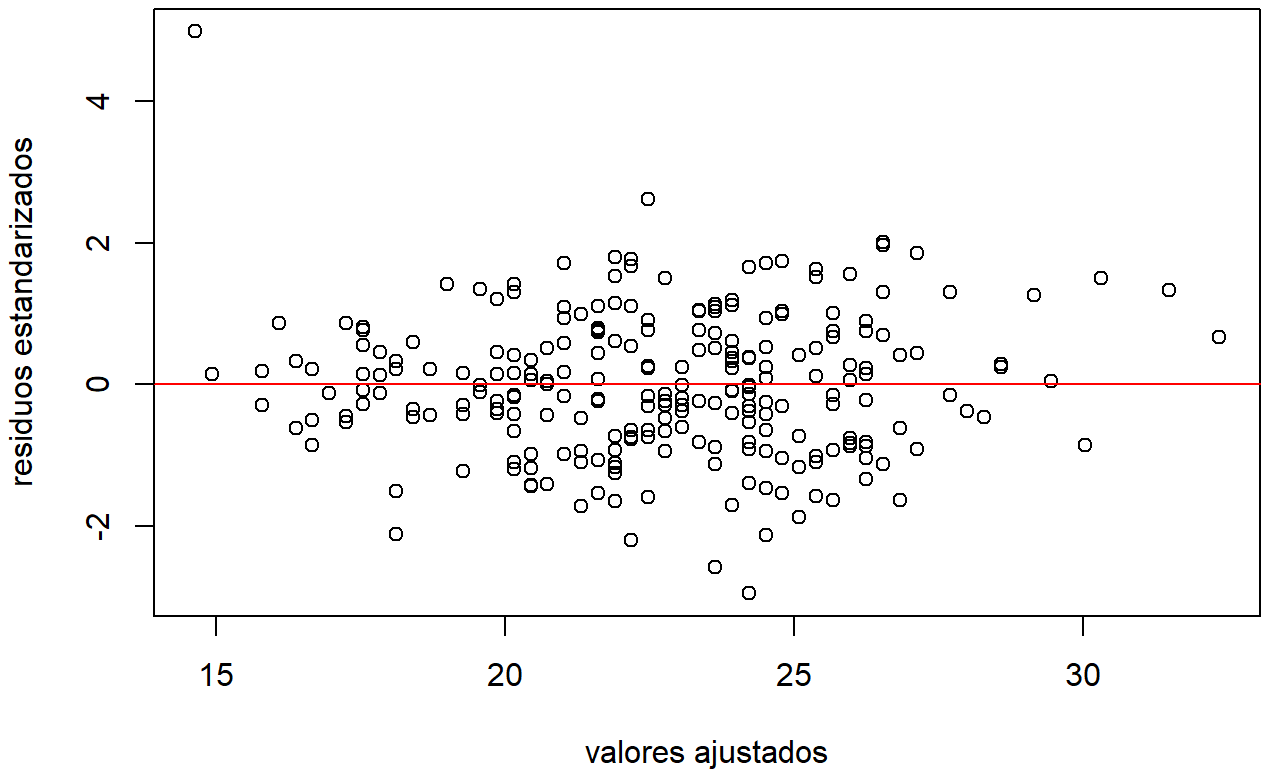
Normalidad de los errores

```
res<- rstandard(MLS)
par(mfrow=c(1,3))
hist(res)
boxplot(res)
qqnorm(res, pch=19,col = "gray50")
qqline(res)
```



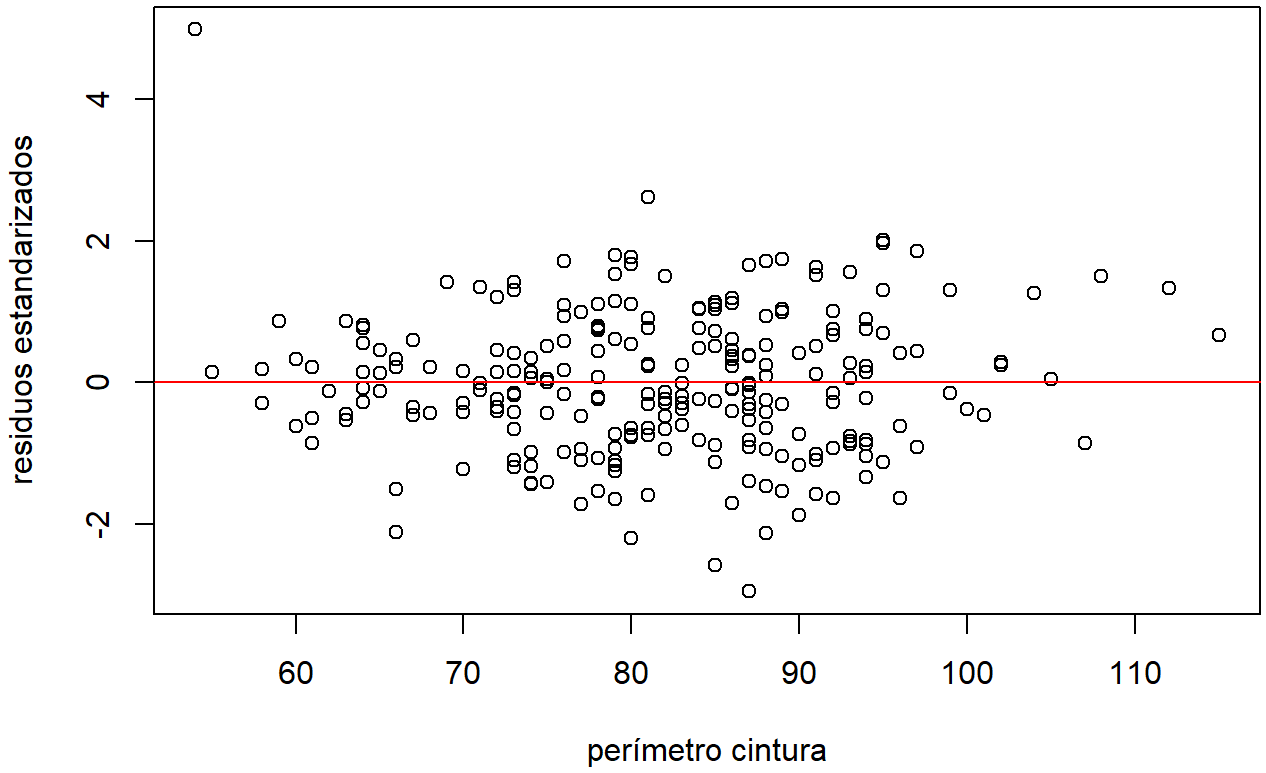
Varianza de los errores(heterocedasticidad)

```
plot(fitted.values(MLS), rstandard(MLS), xlab="valores ajustados",
ylab="residuos estandarizados")
abline(h=0, col="red")
```



Valores atípicos

```
plot(datosdf2$WC, rstandard(MLS),xlab = "perímetro cintura", ylab="residuos
estandarizados")
abline(h=0, col="red")
```



Los residuos parecen normales y la varianza es bastante constante. Hay algunos valores que presentan mucha dispersión.

Modelo lineal múltiple.

Primero planteamos el modelo completo, con la resistencia a la insulina como variable predictora:

```
LMC<- lm(HOMA_IR~ BMI + WC + TG+ TC+ FBS..mmol.l.+SBP+DBP+
F.Ins.pmol.L.+HOMA_B+ Adiponectin.microgm.ml...5000.X. +Leptin..ng.ml.+
Body.Fat....from.Age..BMI +DM_status,data = datosdf2)
summary(LMC)
##
## Call:
## lm(formula = HOMA_IR ~ BMI + WC + TG + TC + FBS..mmol.l. + SBP +
##     DBP + F.Ins.pmol.L. + HOMA_B + Adiponectin.microgm.ml...5000.X. +
##     Leptin..ng.ml. + Body.Fat....from.Age..BMI + DM_status, data =
datosdf2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2645 -0.2722 -0.0112  0.2285  6.8104
##
```

```
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       -0.9653414  0.4405707  -2.191 0.029409 *
## BMI                               -0.0055043  0.0269372  -0.204 0.838263
## WC                                -0.0044927  0.0074047  -0.607 0.544601
## TG                                -0.0004319  0.0004997  -0.864 0.388311
## TC                                -0.0013248  0.0011797  -1.123 0.262561
## FBS..mmol.l.                       0.3147131  0.0329641   9.547  < 2e-16
***
## SBP                               -0.0020398  0.0025222  -0.809 0.419485
## DBP                                0.0049879  0.0043412   1.149 0.251712
## F.Ins.pmol.L.                      0.0533522  0.0024322  21.936  < 2e-16
***
## HOMA_B                            -0.0044100  0.0011229  -3.928 0.000112
***
## Adiponectin.microgm.ml...5000.X. -0.0054536  0.0146218  -0.373 0.709495
## Leptin..ng.ml.                     0.0010610  0.0016850   0.630 0.529514
## Body.Fat....from.Age..BMI         -0.0091624  0.0094716  -0.967 0.334345
## DM_statusNDM                      -0.6421084  0.2419725  -2.654 0.008497 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6457 on 239 degrees of freedom
## Multiple R-squared:  0.8448, Adjusted R-squared:  0.8364
## F-statistic: 100.1 on 13 and 239 DF,  p-value: < 2.2e-16
```

Por los valores de R-squared, vemos que modelo se ajusta bastante bien.

(7) ANOVA (1p)

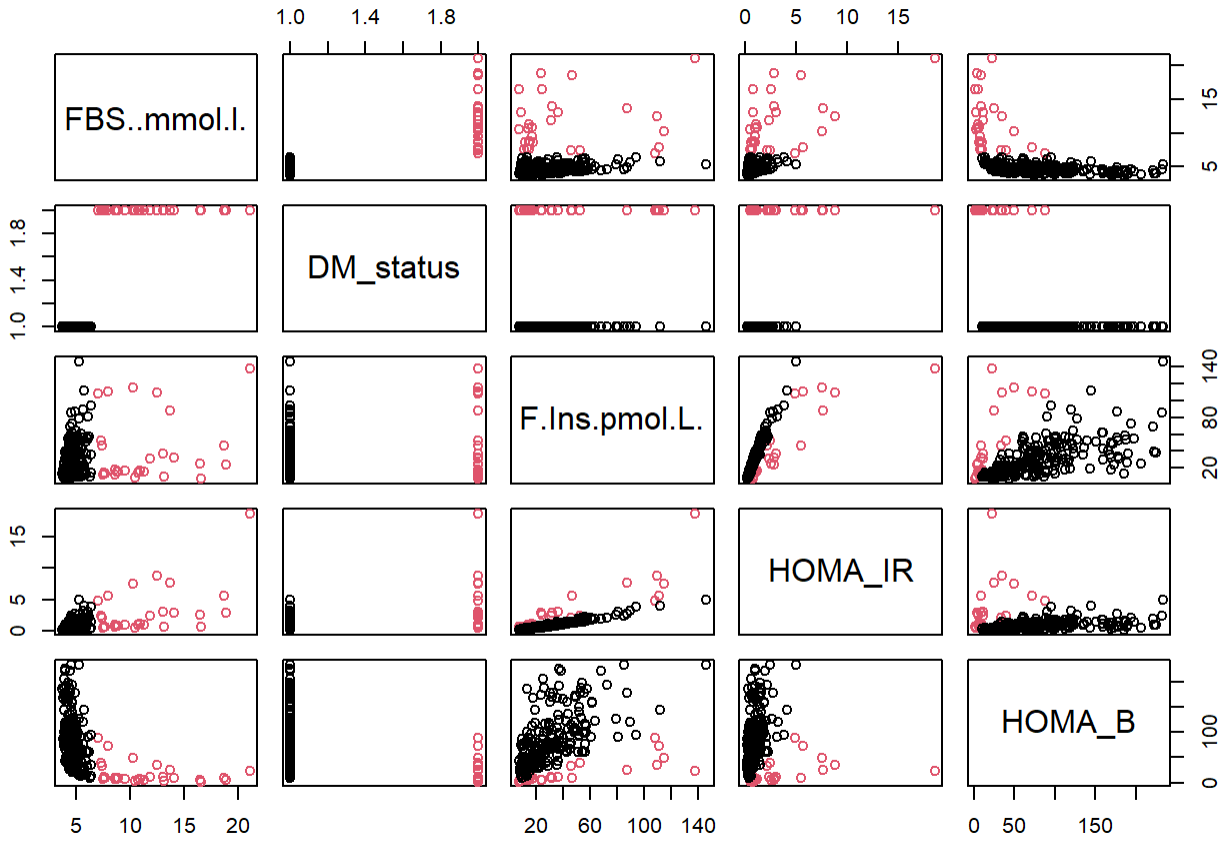
Utilizamos las variables más significativas para crear un nuevo modelo simplificado:

```
LMC2<- lm(HOMA_IR~ FBS..mmol.l.+ F.Ins.pmol.L.+HOMA_B+ DM_status,data =
datosdf2)
summary(LMC2)
##
## Call:
## lm(formula = HOMA_IR ~ FBS..mmol.l. + F.Ins.pmol.L. + HOMA_B +
##     DM_status, data = datosdf2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3757 -0.2569  0.0133  0.2069  7.3735
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.806974   0.179937 -10.042   <2e-16 ***
## FBS..mmol.l.   0.323636   0.032510   9.955   <2e-16 ***
## F.Ins.pmol.L.  0.051332   0.002342  21.916   <2e-16 ***
## HOMA_B        -0.004062   0.001108  -3.667   0.0003 ***
## DM_statusNDM  -0.760259   0.238272  -3.191   0.0016 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6523 on 248 degrees of freedom
## Multiple R-squared:  0.8356, Adjusted R-squared:  0.833
```

```
## F-statistic: 315.2 on 4 and 248 DF,  p-value: < 2.2e-16
anova(LMC, LMC2)
## Analysis of Variance Table
##
## Model 1: HOMA_IR ~ BMI + WC + TG + TC + FBS..mmol.l. + SBP + DBP +
F.Ins.pmol.L. +
##     HOMA_B + Adiponectin.microgm.ml...5000.X. + Leptin..ng.ml. +
##     Body.Fat....from.Age..BMI + DM_status
## Model 2: HOMA_IR ~ FBS..mmol.l. + F.Ins.pmol.L. + HOMA_B + DM_status
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1    239  99.635
## 2    248 105.533 -9   -5.8981 1.572 0.1243
```

Por los resultados obtenidos en Anova y los valores de R-squared, nos quedamos con este segundo modelo simplificado.

```
pairs(datosdf2[,c(6,11,12,22,23)], col=datosdf2$DM_status)
```



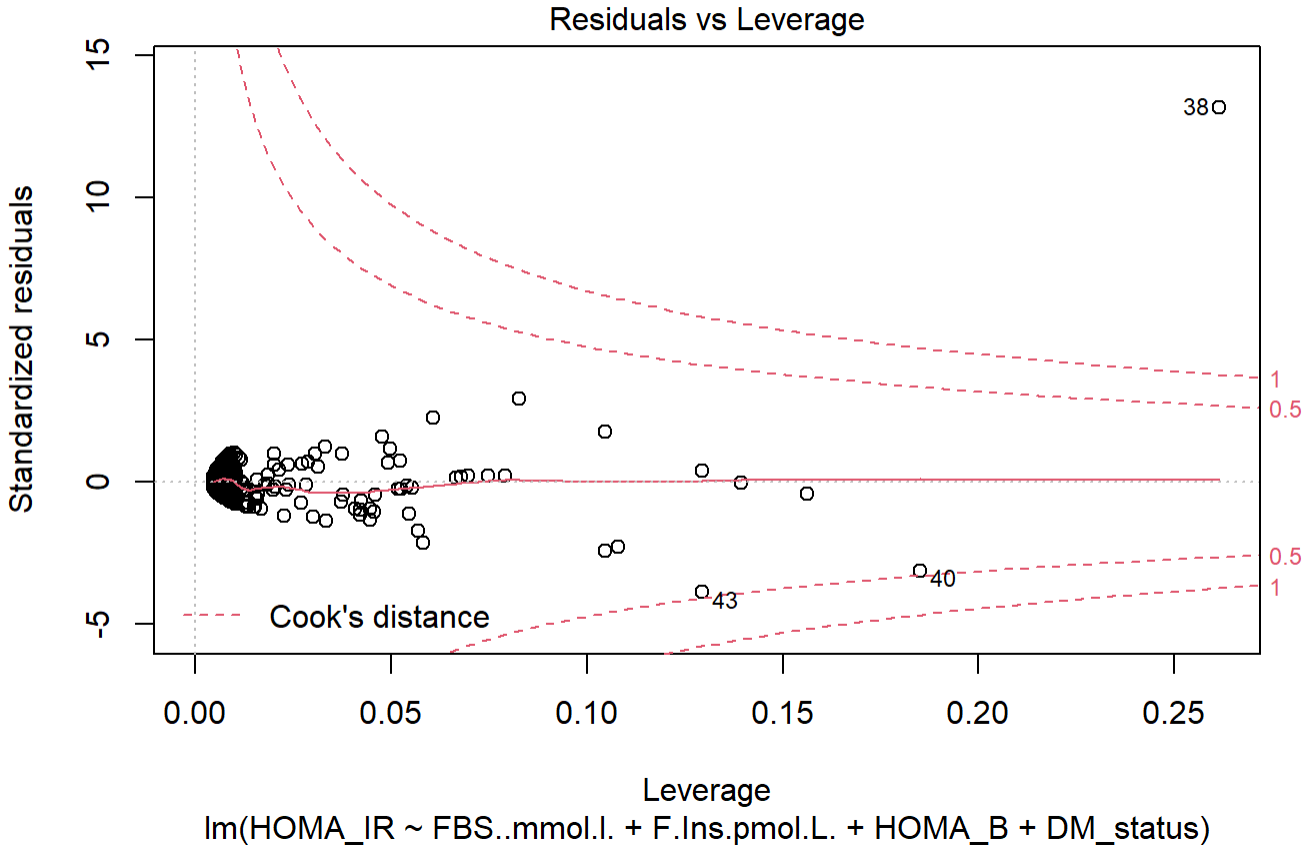
Intervalos de confianza para los coeficientes

```
confint(LMC2)
##                       2.5 %       97.5 %
## (Intercept)    -2.161373403 -1.452575457
## FBS..mmol.l.    0.259605436  0.387666427
```

```
## F.Ins.pmol.L.  0.046718443  0.055944881
## HOMA_B        -0.006244309 -0.001880617
## DM_statusNDM  -1.229554129 -0.290963696
```

Diagnosis: normalidad, heterocedasticidad.

```
plot(LMC2)
```



Los gráficos primero y tercero se utilizan para contrastar gráficamente la independencia, la homocedasticidad y la linealidad de los residuos. Idealmente, los residuos deben estar aleatoriamente distribuidos a lo largo del gráfico, sin formar ningún tipo de patrón.

El gráfico Q- Q, por su parte, se utiliza para contrastar la normalidad de los residuos. Lo deseable es que los residuos estandarizados estén lo más cerca posible a la línea punteada que aparece en el gráfico.

El gráfico de leverages frente a los residuos estandarizados se utiliza para detectar puntos con una influencia importante en el cálculo de las estimaciones de los parámetros. En caso de detectarse algún punto fuera de los límites que establecen las líneas discontinuas debe estudiarse este punto de forma aislada para detectar, por ejemplo, si la elevada importancia de esa observación se debe a un error.

Test de normalidad: Shapiro Wilk

```
shapiro.test(residuals(LMC2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(LMC2)
## W = 0.62965, p-value < 2.2e-16
```

Por el valor de p, rechazamos la hipótesis de normalidad de los residuos También podemo realizar el test de Kolmogorov-Smirnov:

```
ks.test(LMC2$residuals, "pnorm")
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  LMC2$residuals
## D = 0.2365, p-value = 1.022e-12
## alternative hypothesis: two-sided
```

Los resultados del test nos confirman lo que se intuía en el gráfico Q-Q: a un 5% de significación los residuos no siguen una distribución normal, puesto que el p-valor que se obtiene es menor que 0.05.

Heterocedasticidad: Varianza no constante de los residuos.

Test de puntuación de varianza no constante, y test de Breusch-Pagan:

```
ncvTest(LMC2)
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 2170.308, Df = 1, p = < 2.22e-16
bptest(LMC2)
##
##  studentized Breusch-Pagan test
##
## data:  LMC2
## BP = 87.007, df = 4, p-value < 2.2e-16
```

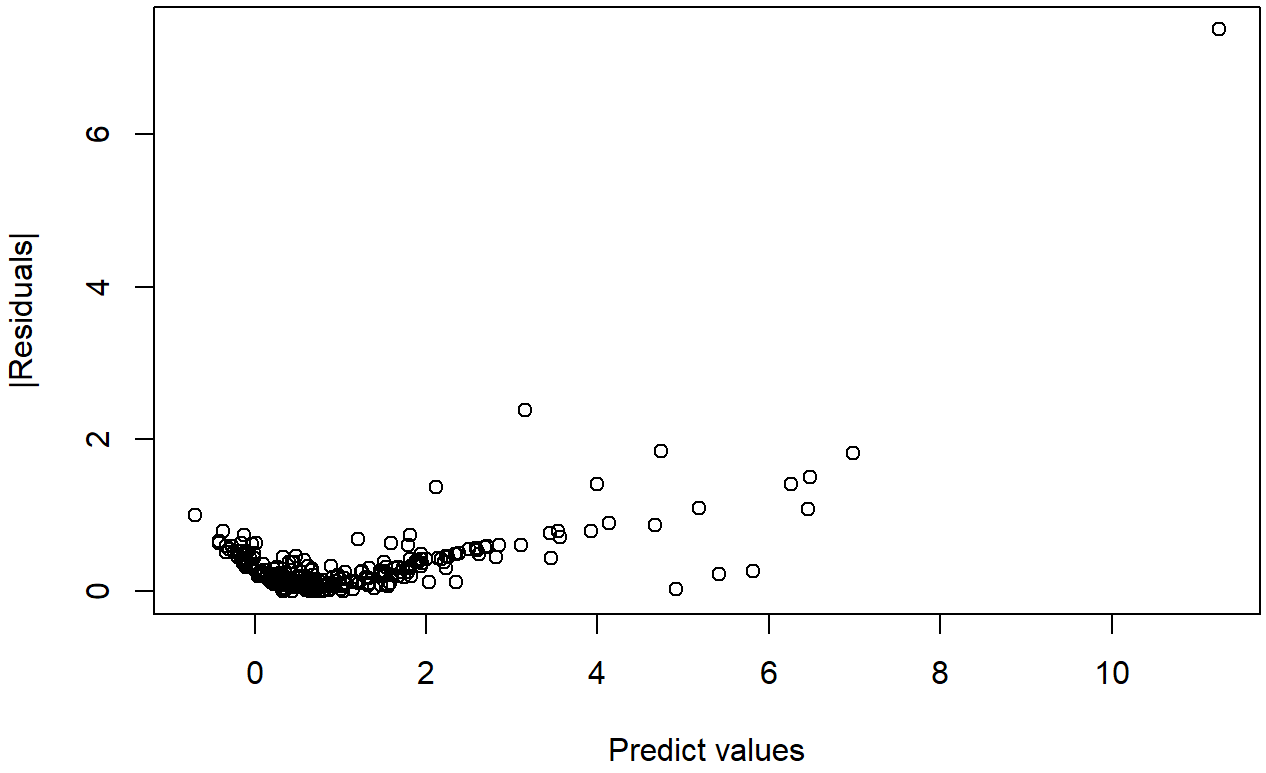
Test de Durbin Watson:

```
dwtest(LMC2)
##
##  Durbin-Watson test
##
## data:  LMC2
## DW = 1.3525, p-value = 5.103e-08
## alternative hypothesis: true autocorrelation is greater than 0
```

De nuevo, rechazamos la hipótesis de que los residuos son independientes.

Otra manera de comprobarlo: ajustando una recta a los valores absolutos de los residuos (o sus raíces cuadradas) y los valores predichos:

```
plot(fitted(LMC2),abs(residuals(LMC2)),xlab="Predict
values",ylab="|Residuals|")
```

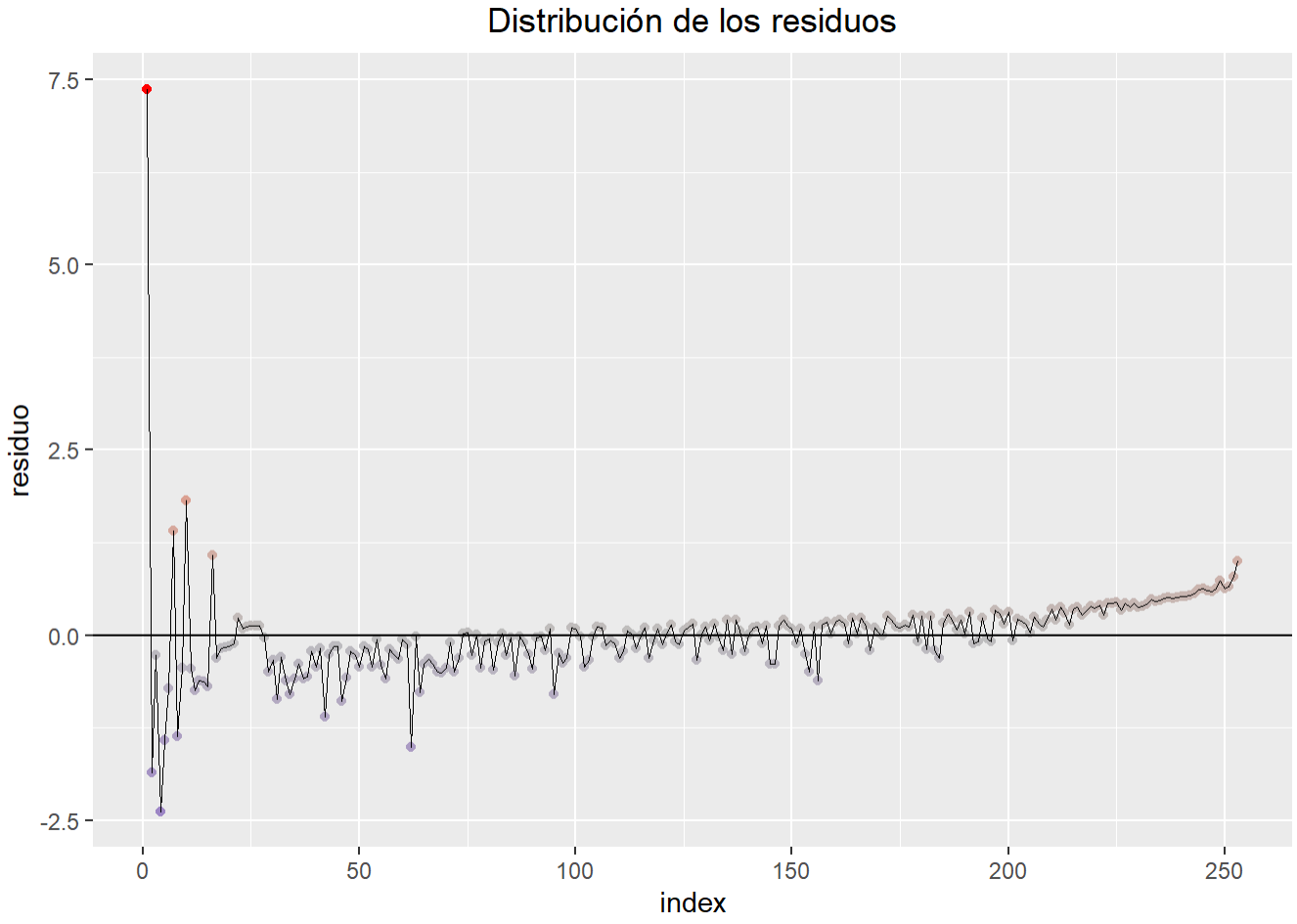



```
summary(lm(sqrt(abs(residuals(LMC2)))~fitted(LMC2)))
##
## Call:
## lm(formula = sqrt(abs(residuals(LMC2))) ~ fitted(LMC2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.74769 -0.13531 -0.01470  0.09822  1.07273
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.384954   0.018634   20.66   <2e-16 ***
## fitted(LMC2) 0.111805   0.009975   11.21   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2311 on 251 degrees of freedom
## Multiple R-squared:  0.3335, Adjusted R-squared:  0.3309
## F-statistic: 125.6 on 1 and 251 DF,  p-value: < 2.2e-16
```

** Análisis gráfico autocorrelación de residuos:**

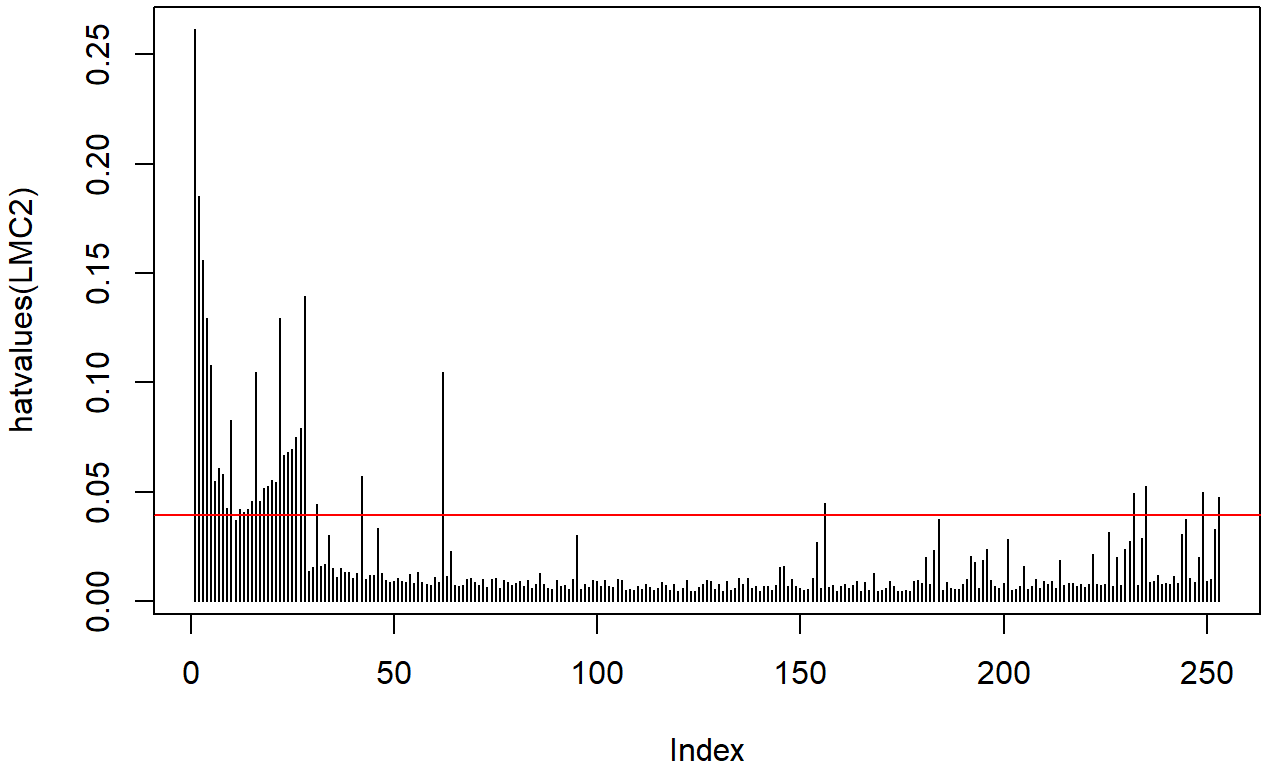
```
ggplot(data = datosdf2, aes(x = seq_along(LMC2$residuals),
```

```
                        y = LMC2$residuals)) +
geom_point(aes(color = LMC2$residuals)) +
scale_color_gradient2(low = "blue3", mid = "grey", high = "red") +
geom_line(size = 0.3) +
labs(title = "Distribución de los residuos", x = "index", y = "residuo")+
geom_hline(yintercept = 0) +
theme(plot.title = element_text(hjust = 0.5), legend.position = "none")
```



Leverage

```
plot(hatvalues(LMC2), type="h")
p<- length(LMC2$coefficients)
n<- length(LMC2$fitted.values)
cutoff<- 2*p/n
abline(h=cutoff, col="red")
```




```
which(hatvalues(LMC2)>cutoff)
##  38  40  41  43  44  46  47  53  54  58  65  68  71  73  74  78  82  83
85  86
##   1   2   3   4   5   6   7   8   9  10  12  13  14  15  16  17  18  19
20  21
##  91  98 102 104 107 108 113 122 150 181 332 456 462 477 482
##  22  23  24  25  26  27  28  31  42  62 156 232 235 249 253
head(sort(hatvalues(LMC2), decreasing = T))
##        38        40        41       113        43        91
## 0.2613593 0.1851367 0.1561394 0.1394024 0.1294910 0.1294337
```

Los valores 38, 40, 41, 113, 43, y 91 son los que tienen un leverage más alto.

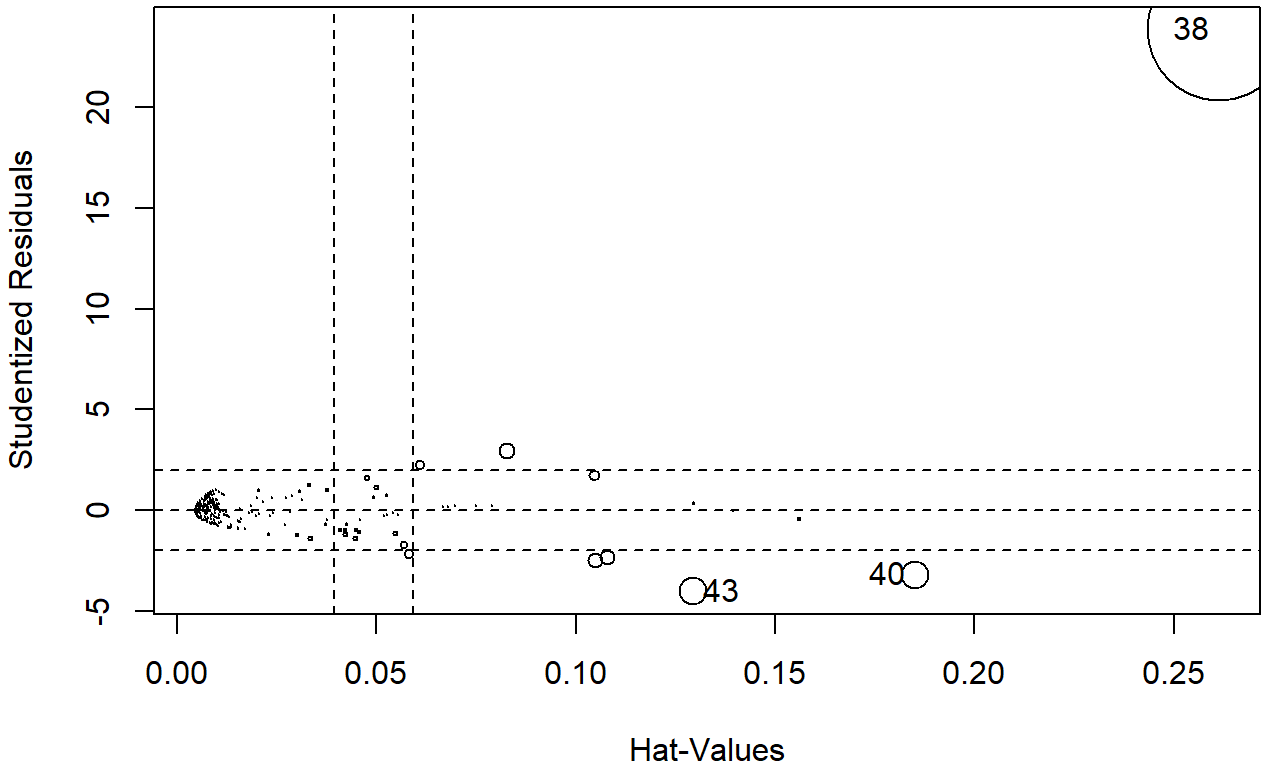
Influencia:

```
head(sort(cooks.distance(LMC2), decreasing = T))
##         38         43         40         58        181         44
## 12.2410527  0.4532648  0.4474035  0.1522983  0.1393824  0.1268850
```

Los valores 38,43 y 40 son los que tienen mayor influencia.

```
summary(influence.measures(model=LMC2))
## Potentially influential observations of
##   lm(formula = HOMA_IR ~ FBS..mmol.l. + F.Ins.pmol.L. + HOMA_B +
DM_status, data = datosdf2) :
```

```
##
##     dfb.1_  dfb.FBS. dfb.F.I. dfb.HOMA dfb.DM_N dffit   cov.r   cook.d
hat
## 38  -9.59_*  9.77_*   4.53_*  -0.98    -7.01_*  14.20_*  0.00_* 12.24_*
0.26_*
## 40   1.17_* -1.36_*   0.66    -0.55     0.67    -1.52_*  1.02    0.45
0.19_*
## 41   0.15   -0.17     0.05    -0.05     0.09    -0.19    1.20_*  0.01
0.16_*
## 43   1.04_* -1.25_*   0.86    -0.63     0.45    -1.55_*  0.85_*  0.45
0.13_*
## 44   0.55   -0.65     0.34    -0.27     0.24    -0.80_*  1.03    0.13
0.11_*
## 47  -0.15    0.13     0.27    -0.13     0.05     0.57_*  0.98    0.06
0.06_*
## 53   0.20   -0.26     0.29    -0.18    -0.06    -0.54_*  0.99    0.06
0.06
## 58  -0.01   -0.09     0.64    -0.34     0.21     0.89_*  0.94_*  0.15
0.08_*
## 74   0.17   -0.26     0.48    -0.26     0.29     0.60_*  1.07_*  0.07
0.10_*
## 78  -0.04    0.04     0.02     0.00    -0.08    -0.11    1.06_*  0.00
0.05
## 82  -0.03    0.03     0.01     0.00    -0.05    -0.06    1.07_*  0.00
0.05
## 83  -0.03    0.03     0.01     0.00    -0.05    -0.06    1.08_*  0.00
0.05
## 85  -0.03    0.03     0.01     0.00    -0.05    -0.05    1.08_*  0.00
0.06
## 86  -0.02    0.02     0.00     0.00    -0.03    -0.04    1.08_*  0.00
0.05
## 91   0.07   -0.09     0.10    -0.05     0.10     0.14    1.17_*  0.00
0.13_*
## 98   0.02   -0.02     0.00     0.00     0.03     0.04    1.09_*  0.00
0.07_*
## 102  0.03   -0.03     0.00     0.00     0.04     0.05    1.09_*  0.00
0.07_*
## 104  0.03   -0.03     0.00     0.00     0.05     0.06    1.10_*  0.00
0.07_*
## 107  0.04   -0.04     0.02    -0.01     0.05     0.06    1.10_*  0.00
0.07_*
## 108  0.03   -0.04     0.02    -0.01     0.05     0.06    1.11_*  0.00
0.08_*
## 113 -0.01    0.02    -0.02     0.01    -0.02    -0.02    1.19_*  0.00
0.14_*
## 150  0.03    0.03    -0.37     0.11     0.06    -0.43_*  1.02    0.04
0.06
## 181  0.14    0.08    -0.63    -0.01     0.03    -0.84_*  1.01    0.14
0.10_*
## 456 -0.05    0.03    -0.07     0.14     0.01     0.15    1.06_*  0.00
0.05
## 462 -0.06    0.03    -0.09     0.16     0.02     0.17    1.07_*  0.01
0.05
influencePlot(model=LMC2)
```




```
##       StudRes       Hat      CookD
## 38 23.863726 0.2613593 12.2410527
## 40 -3.195585 0.1851367  0.4474035
## 43 -4.020851 0.1294910  0.4532648
```

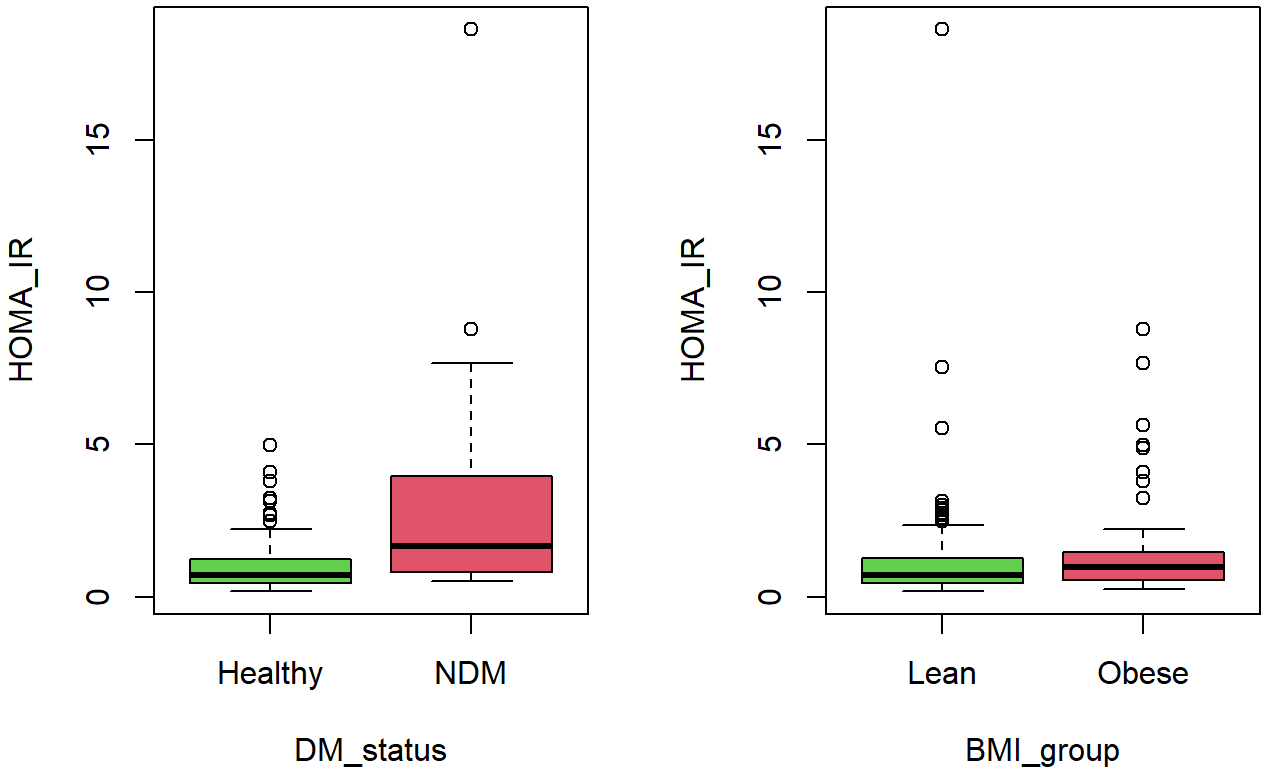
Algunas observaciones tienen un residuo estandarizado absoluto próximo a 3 (1.73 si se considera la raíz cuadrada) lo que es indicativo de observación atípica. Valores de Leverages (hat) mayores que 2x((p+1)/n), siendo p el número de predictores y n el número de observaciones, o valores de Cook mayores de 1 se consideran influyentes. Todo ello reduce en gran medida la robustez de la estimación del error estándar de los coeficientes de correlación estimados y con ello la del modelo es su conjunto.

** Hipótesis de 2 modelos**

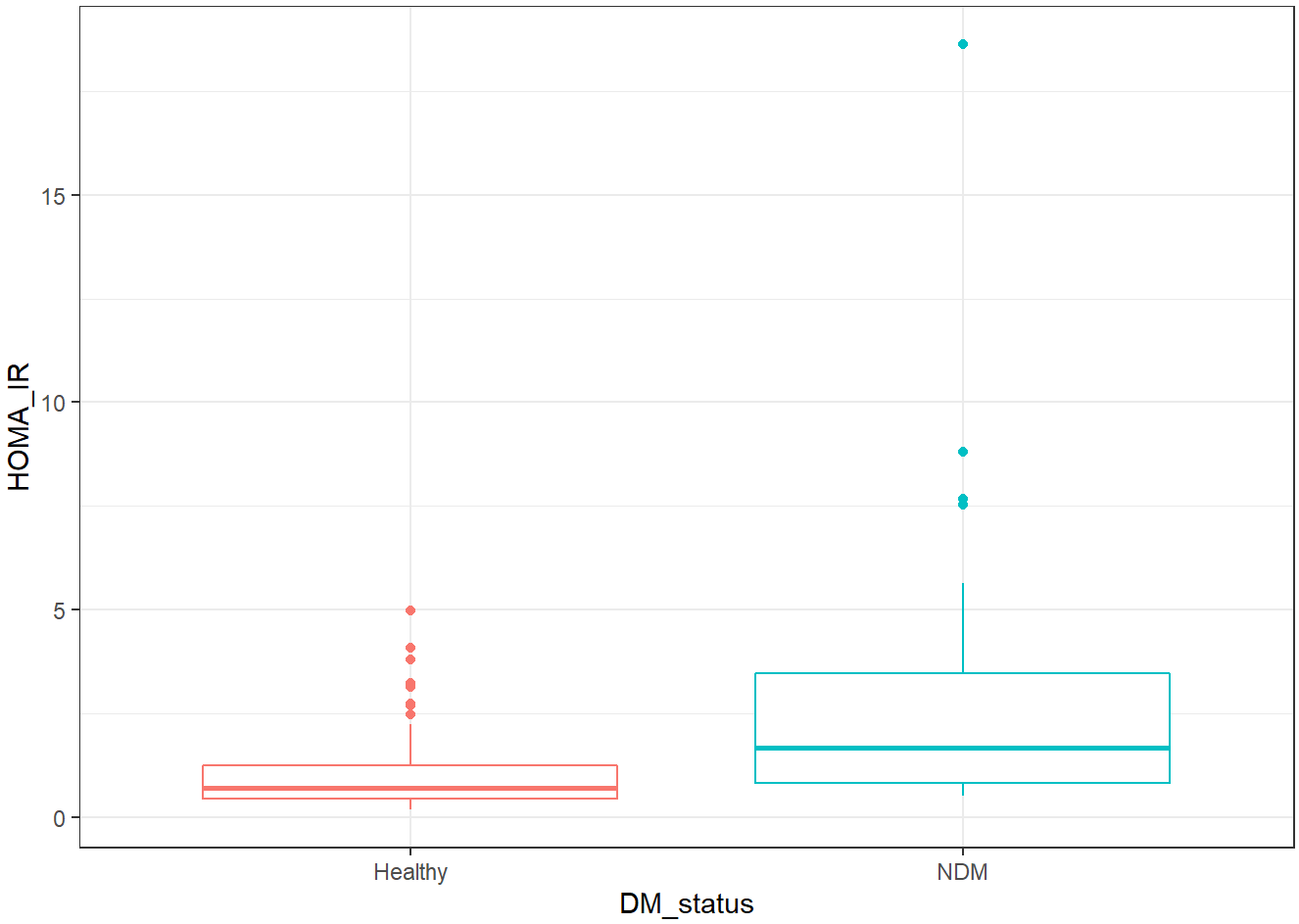
¿Depende la resistencia a la insulina de ser obeso? y de ser diabético tipo 2?

Uilizaremos HOMA_IR como predictor, para ver si existen diferencias en función de ser o no diabético, y de ser obeso o no.

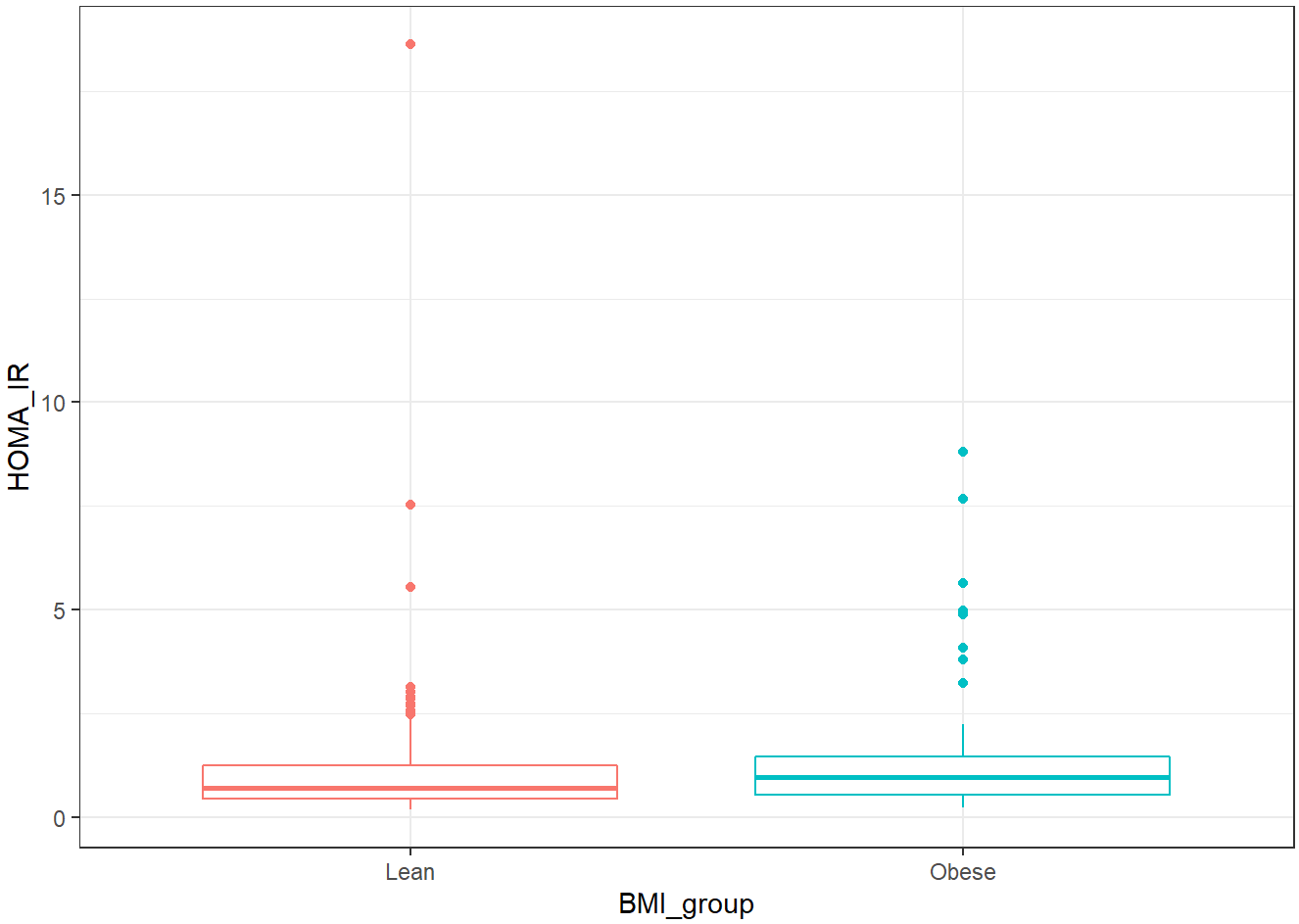
```
par(mfrow=c(1,2))
boxplot(HOMA_IR~ DM_status, data=datosdf2, col=c(3,2))
boxplot(HOMA_IR~BMI_group, data = datosdf2, col=c(3,2))
```




```
ggplot(data = datosdf2) +
  geom_boxplot(aes(x = DM_status, y = HOMA_IR, colour = DM_status)) +
  theme_bw() + theme(legend.position = "none")
```

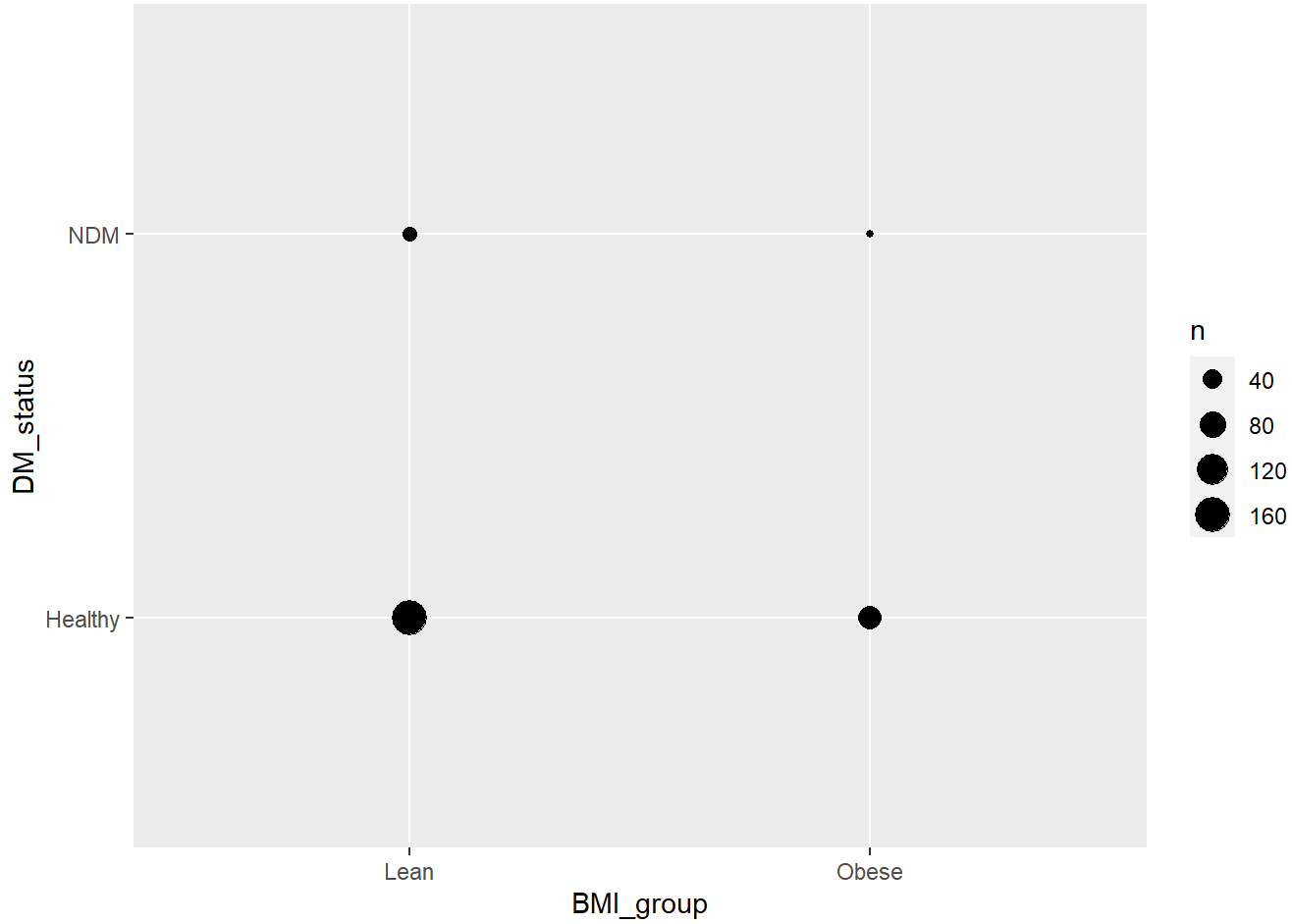



```
ggplot(data = datosdf2) +
  geom_boxplot(aes(x = BMI_group, y = HOMA_IR, colour = BMI_group)) +
  theme_bw() + theme(legend.position = "none")
```



Test de Fisher. Para saber si existe relación entre dos variables categóricas, podemos hacer un test de Fisher a partir de una tabla de contingencia:

```
tabla <- table(datosdf2$BMI_group, datosdf2$DM_status, dnn = c("obeso o no",
"diabético o no"))
ggplot(data = datosdf2) +
  geom_count(mapping = aes(x = BMI_group, y = DM_status))
```



Ho : Las variables son independientes por lo que una variable no varía entre los distintos niveles de la otra variable. Ha: Las variables son dependientes, una variable varía entre los distintos niveles de la otra variable.

```
fisher.test(x = tabla, alternative = "two.sided")
##
##  Fisher's Exact Test for Count Data
##
## data:  tabla
## p-value = 0.5094
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.4903449 3.2198846
## sample estimates:
## odds ratio
##   1.301234
```

No podríamos rechazar la hipótesis nula (alpha=0.05). Dado que el test de Fisher contrasta si las variables están relacionadas, al tamaño del efecto se le conoce como fuerza de asociación. Existen múltiples medidas de asociación, entre las que destacan phi o Cramer's V. Los límites empleados para su clasificación son: pequeño: 0.1 mediano: 0.3 grande: 0.5 En R se pueden calcular mediante la función assocstats() del paquete vcd.

```
assocstats(x = tabla)
```

```
##                       X^2 df P(> X^2)
## Likelihood Ratio 0.36547  1  0.54548
## Pearson          0.37649  1  0.53949
##
## Phi-Coefficient   : 0.039
## Contingency Coeff.: 0.039
## Cramer's V        : 0.039
```

En este ejemplo no se satisface la condición de frecuencias marginales fijas y por lo tanto el test de Fisher no es exacto. Aun así, parece ser que las dos variables no están relacionadas. El tamaño de la fuerza de asociación (tamaño de efecto) cuantificado por phi o Cramer's V es grande.

χ2 de Pearson (test de independencia)

Es el test aproximado equivalente a su versión exacta test de Fisher. Debido a los requerimientos de cálculo del test de Fisher, cuando hay muchas observaciones o muchos niveles, se emplea el test χ2 de independencia. La distribución chi-cuadrado es siempre positiva, por lo que para calcular el p-value solo se tiene en cuenta la cola superior.

```
chisq.test(x = tabla)
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabla
## X-squared = 0.15101, df = 1, p-value = 0.6976
```

Solución mediante simulación:

```
chisq.test(tabla, simulate.p.value = TRUE, B = 5000)
##
##  Pearson's Chi-squared test with simulated p-value (based on 5000
##  replicates)
##
## data:  tabla
## X-squared = 0.37649, df = NA, p-value = 0.6501
```

Anova.

Contrastamos si la resistencia a la insulina es igual en diabético y no diabéticos, y en obesos o no:

```
anova(lm(HOMA_IR~DM_status, data = datosdf2))
## Analysis of Variance Table
##
## Response: HOMA_IR
##            Df Sum Sq Mean Sq F value    Pr(>F)
## DM_status   1 124.94  124.94  60.641 1.802e-13 ***
## Residuals 251 517.12    2.06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
anova(lm(HOMA_IR~BMI_group, data = datosdf2))
## Analysis of Variance Table
##
## Response: HOMA_IR
##            Df Sum Sq Mean Sq F value Pr(>F)
## BMI_group   1   6.87  6.8668  2.7134 0.1008
## Residuals 251 635.19  2.5307
```

Los valores de HOMA_IR dependen de la condición de estar o no enfermo, pero no del IMC

(nivel de significación 0.05)

Se podría estudiar también como influyen las variables en la resistencia a la insulina, pero específicamente en individuos diabéticos y/o sanos independientemente, de esta manera obtenemos valores de t para cada grupo

```
datHealth<- subset(datosdf2, datosdf2$DM_status=="Healthy")
datDM <- subset(datosdf2, datosdf2$DM_status=="NDM")
lmH<- lm(HOMA_IR~ BMI + WC + TG+ TC+ FBS..mmol.l.+SBP+DBP+
F.Ins.pmol.L.+HOMA_B+ Adiponectin.microgm.ml...5000.X. +Leptin..ng.ml.,data =
datHealth)
lmDM<-lm(HOMA_IR~ BMI + WC + TG+ TC+ FBS..mmol.l.+SBP+DBP+
F.Ins.pmol.L.+HOMA_B+ Adiponectin.microgm.ml...5000.X. +Leptin..ng.ml., data
= datDM)
summary(lmH);summary(lmDM)
##
## Call:
## lm(formula = HOMA_IR ~ BMI + WC + TG + TC + FBS..mmol.l. + SBP +
##     DBP + F.Ins.pmol.L. + HOMA_B + Adiponectin.microgm.ml...5000.X. +
##     Leptin..ng.ml., data = datHealth)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.12364 -0.03763 -0.01393  0.02436  0.42159
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -4.669e-01  7.796e-02  -5.990 8.85e-09
***
## BMI                              3.070e-03  2.155e-03   1.425 0.155748
## WC                               5.676e-04  7.773e-04   0.730 0.466052
## TG                              -6.003e-06  6.018e-05  -0.100 0.920630
## TC                              -1.686e-04  1.362e-04  -1.238 0.217181
## FBS..mmol.l.                     8.222e-02  1.519e-02   5.413 1.66e-07
***
## SBP                              1.709e-04  2.769e-04   0.617 0.537738
## DBP                             -2.520e-04  4.877e-04  -0.517 0.605953
## F.Ins.pmol.L.                    3.597e-02  4.968e-04  72.402  < 2e-16
***
## HOMA_B                          -1.707e-03  2.010e-04  -8.492 3.49e-15
***
## Adiponectin.microgm.ml...5000.X. -8.638e-04  1.673e-03  -0.516 0.606255
## Leptin..ng.ml.                  -6.762e-04  1.908e-04  -3.543 0.000486
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0693 on 213 degrees of freedom
## Multiple R-squared:  0.9909, Adjusted R-squared:  0.9904
## F-statistic:  2103 on 11 and 213 DF,  p-value: < 2.2e-16
##
## Call:
## lm(formula = HOMA_IR ~ BMI + WC + TG + TC + FBS..mmol.l. + SBP +
##     DBP + F.Ins.pmol.L. + HOMA_B + Adiponectin.microgm.ml...5000.X. +
##     Leptin..ng.ml., data = datDM)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.32848 -0.31294 -0.01563  0.23274  1.47021
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       0.284059   2.889828   0.098 0.922918
## BMI                               0.004469   0.106880   0.042 0.967167
## WC                               -0.015137   0.038515  -0.393 0.699501
## TG                               -0.000643   0.001731  -0.371 0.715220
## TC                               -0.003825   0.004328  -0.884 0.389879
## FBS..mmol.l.                      0.114273   0.076129   1.501 0.152818
## SBP                               0.004334   0.012629   0.343 0.735925
## DBP                               0.004344   0.023272   0.187 0.854280
## F.Ins.pmol.L.                     0.128317   0.011106  11.553 3.56e-09 ***
## HOMA_B                           -0.101566   0.021683  -4.684 0.000249 ***
## Adiponectin.microgm.ml...5000.X. -0.125664   0.071720  -1.752 0.098890 .
## Leptin..ng.ml.                   -0.004500   0.004913  -0.916 0.373319
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7674 on 16 degrees of freedom
## Multiple R-squared:  0.9767, Adjusted R-squared:  0.9607
## F-statistic: 61.07 on 11 and 16 DF,  p-value: 6.326e-11
```

Por los valores de R-squared, estos modelos explicarían un alto porcentaje de la variabilidad. Observamos que los niveles de leptina son más significativos en el caso de los individuos sanos, en cambio en el caso de los diabéticos es más significativa la adiponectina. Además, vemos que los niveles de azúcar en ayunas, aunque explican algo de variabilidad, no son tan significantes en el caso de los diabéticos, pero sí en el grupo de individuos sanos.

Por lo tanto, estos modelos también se pueden simplificar usando las variables con nivel de significación más alto:

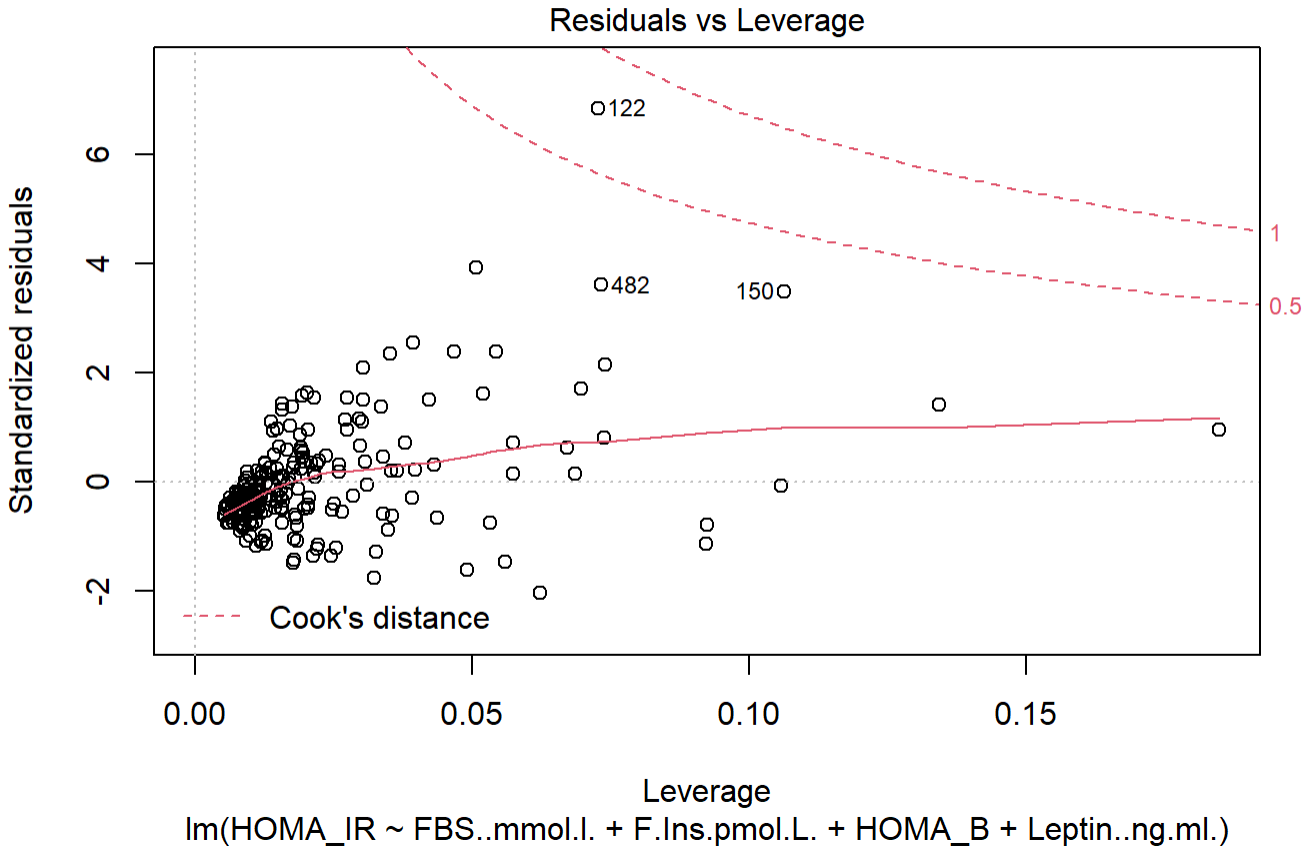
```
lmH2<- lm(HOMA_IR~ FBS..mmol.l.+ F.Ins.pmol.L.+HOMA_B+ Leptin..ng.ml.,data =
datHealth)
lmDM2<-lm(HOMA_IR~ F.Ins.pmol.L.+HOMA_B+ Adiponectin.microgm.ml...5000.X.,
data = datDM)
anova(lmH, lmH2)
## Analysis of Variance Table
##
## Model 1: HOMA_IR ~ BMI + WC + TG + TC + FBS..mmol.l. + SBP + DBP +
F.Ins.pmol.L. +
##     HOMA_B + Adiponectin.microgm.ml...5000.X. + Leptin..ng.ml.
## Model 2: HOMA_IR ~ FBS..mmol.l. + F.Ins.pmol.L. + HOMA_B + Leptin..ng.ml.
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    213 1.0229
## 2    220 1.0813 -7 -0.058344 1.7355 0.1021
anova(lmDM, lmDM2)
## Analysis of Variance Table
##
## Model 1: HOMA_IR ~ BMI + WC + TG + TC + FBS..mmol.l. + SBP + DBP +
F.Ins.pmol.L. +
##     HOMA_B + Adiponectin.microgm.ml...5000.X. + Leptin..ng.ml.
```

```
## Model 2: HOMA_IR ~ F.Ins.pmol.L. + HOMA_B +
Adiponectin.microgm.ml...5000.X.
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1     16  9.4218
## 2     24 13.5518 -8     -4.13 0.8767 0.5556
summary(lmH2); summary(lmDM2)
##
## Call:
## lm(formula = HOMA_IR ~ FBS..mmol.l. + F.Ins.pmol.L. + HOMA_B +
##     Leptin..ng.ml., data = datHealth)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.13823 -0.03795 -0.01779  0.02225  0.46136
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.4138757  0.0731701  -5.656 4.78e-08 ***
## FBS..mmol.l.    0.0874155  0.0149846   5.834 1.92e-08 ***
## F.Ins.pmol.L.   0.0359473  0.0004971  72.309  < 2e-16 ***
## HOMA_B         -0.0016673  0.0002015  -8.274 1.24e-14 ***
## Leptin..ng.ml. -0.0004826  0.0001765  -2.734  0.00676 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07011 on 220 degrees of freedom
## Multiple R-squared:  0.9904, Adjusted R-squared:  0.9902
## F-statistic:  5647 on 4 and 220 DF,  p-value: < 2.2e-16
##
## Call:
## lm(formula = HOMA_IR ~ F.Ins.pmol.L. + HOMA_B +
Adiponectin.microgm.ml...5000.X.,
##     data = datDM)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83962 -0.24976 -0.05204  0.20139  1.98300
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      0.475716   0.322216   1.476   0.1528
## F.Ins.pmol.L.                    0.141150   0.005924  23.825  < 2e-16 ***
## HOMA_B                          -0.133652   0.011094 -12.048 1.15e-11 ***
## Adiponectin.microgm.ml...5000.X. -0.140563   0.052157  -2.695   0.0127 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7514 on 24 degrees of freedom
## Multiple R-squared:  0.9665, Adjusted R-squared:  0.9624
## F-statistic: 231.1 on 3 and 24 DF,  p-value: < 2.2e-16
confint(lmH2); confint(lmDM2)
##                         2.5 %        97.5 %
## (Intercept)    -0.5580797814 -0.2696716455
## FBS..mmol.l.    0.0578837389  0.1169473139
```

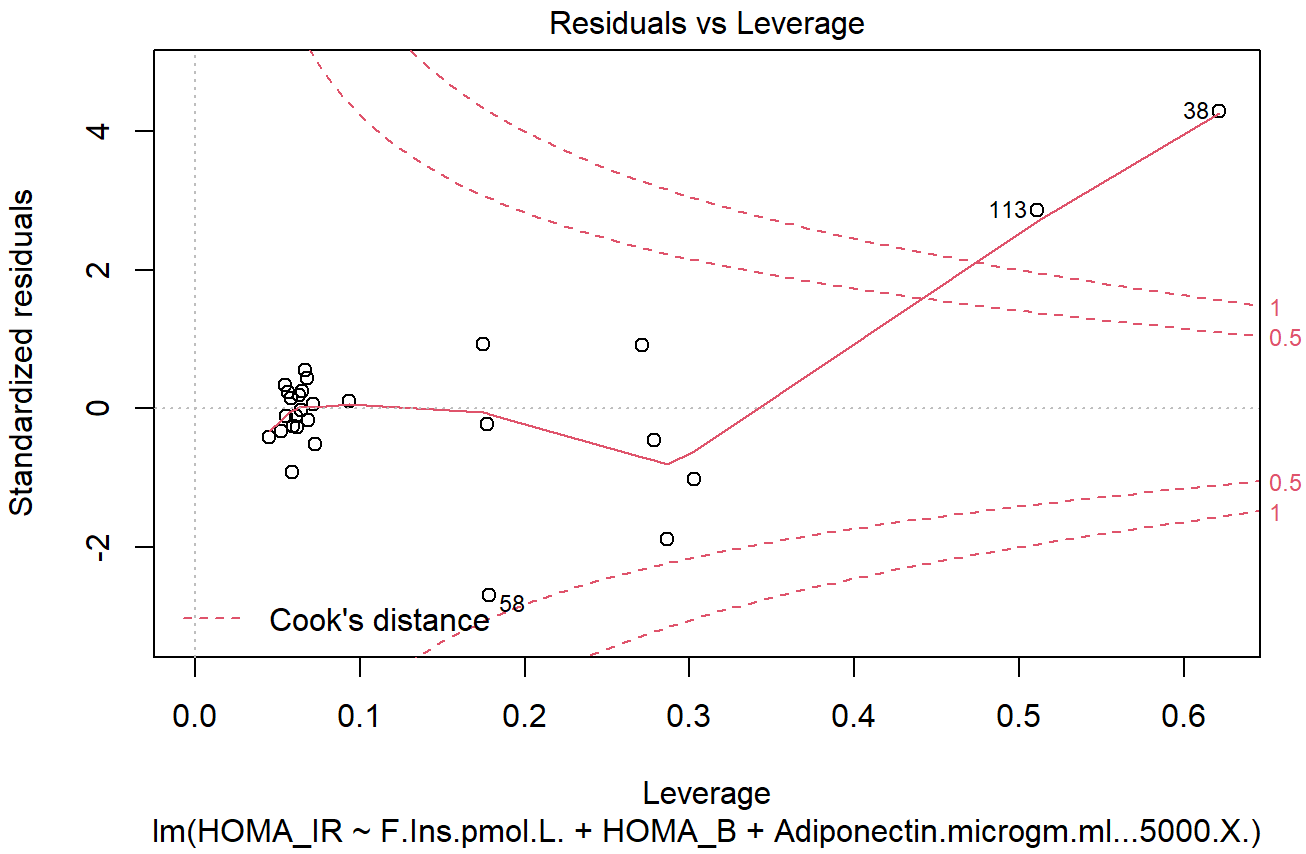
```
## F.Ins.pmol.L.    0.0349675690  0.0369270936
## HOMA_B          -0.0020644454 -0.0012701589
## Leptin..ng.ml. -0.0008305017 -0.0001347466
##                                          2.5 %      97.5 %
## (Intercept)                        -0.1893050  1.14073639
## F.Ins.pmol.L.                       0.1289222  0.15337714
## HOMA_B                             -0.1565487 -0.11075586
## Adiponectin.microgm.ml...5000.X. -0.2482102 -0.03291514
```

De nuevo, nos quedaríamos con el modelo más simple (en ambos grupos).

```
plot(lmH2)
```




```
plot(lmDM2)
```




```
shapiro.test(residuals(lmH2))
##
##  Shapiro-Wilk normality test
##
## data:  residuals(lmH2)
## W = 0.84167, p-value = 2.067e-14
shapiro.test(residuals(lmDM2))
##
##  Shapiro-Wilk normality test
##
## data:  residuals(lmDM2)
## W = 0.90718, p-value = 0.01692
ncvTest(lmH2)
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 185.5233, Df = 1, p = < 2.22e-16
ncvTest(lmDM2)
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 40.16637, Df = 1, p = 2.3323e-10
bptest(lmH2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lmH2
## BP = 34.782, df = 4, p-value = 5.149e-07
bptest(lmDM2)
##
##  studentized Breusch-Pagan test
##
## data:  lmDM2
## BP = 19.236, df = 3, p-value = 0.0002443
```

De nuevo obtenemos resultados de no-normalidad y heterocedasticidad en los residuos.Podría ser útil plantearnos una transformación de los datos, o utilizar un test no parmétrico, como el test de Kruskal-Wallis.

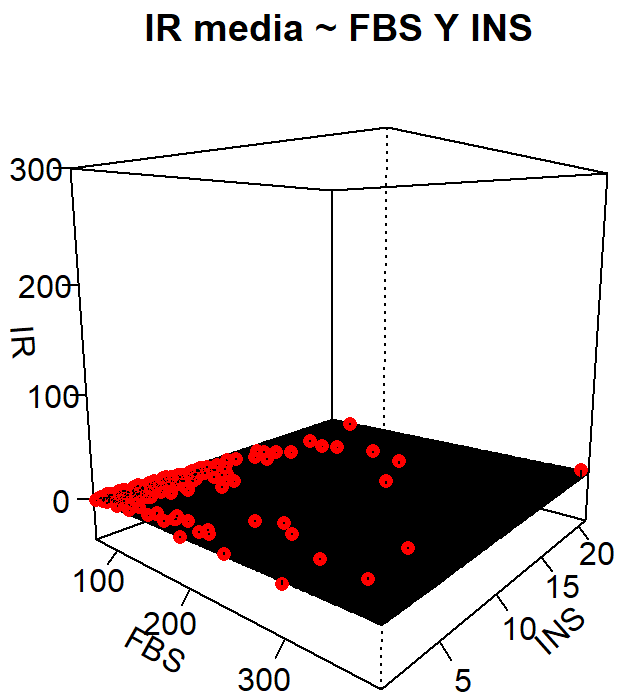
Representación del plano de regresión. Por último, dado un modelo con dos predictores continuos,se puede representar el plano de regresión. Como ejemplo utilizaremos la glucemia e insulinemia como predictores de la IR:

```
modelo<- lm(HOMA_IR~FBS+Ins, data = datosdf2)
rango_fbs <- range(datosdf2$FBS)
nuevos_valores_fbs <- seq(from = rango_fbs[1], to = rango_fbs[2],
                          length.out = 200)
rango_ins <- range(datosdf2$Ins)
nuevos_valores_ins <- seq(from = rango_ins[1], to = rango_ins[2],
                          length.out = 200)

predicciones <- outer(X = nuevos_valores_fbs, Y = nuevos_valores_ins,
                      FUN = function(FBS,Ins) {
                          predict(object = modelo,
                                  newdata = data.frame(FBS,Ins))
                          })

superficie <- persp(x = nuevos_valores_fbs, y = nuevos_valores_ins,
                    z = predicciones,
                    theta = 40, phi = 10,
                    col = "lightblue", shade = 0.1,
                    zlim = range(-40,300),
                    xlab = "FBS", ylab = "INS", zlab = "IR",
                    ticktype = "detailed",
                    main = "IR media ~ FBS Y INS"
                   )

observaciones <- trans3d(datosdf2$FBS, datosdf2$Ins, datosdf2$HOMA_IR,
superficie)
error <- trans3d(datosdf2$FBS, datosdf2$Ins, fitted(modelo), superficie)
points(observaciones, col = "red", pch = 16)
segments(observaciones$x, observaciones$y, error$x, error$y)
```



(8) Conclusiones (1p)

1. El IMC sigue una relación lineal positiva con el perímetro de la cintura.
2. El IMC no parece influir directamente en ser resistente a la insulina. Hemos visto que no hay diferencias significativas entre los grupos obeso/no obeso en general, pero podríamos analizar también si existen diferencias entre obesos y no dentro de cada grupo DM2 no-DM2. En cambio sí que la resistencia a la insulina difiere entre diabéticos o no diabéticos.
3. Los parámetros que más influyen en ser resistente son: para ambos grupos, niveles de insulina en sangre en ayunas y la función de las células beta (de manera negativa, como era de esperar). Observamos que los niveles de leptina son más significativos en el caso de los individuos sanos, en cambio en el caso de los diabéticos es más significativa la adiponectina, ambos de manera negativa también. Los niveles de azúcar en ayunas, aunque explican algo de variabilidad, no son tan significativos en el caso de los diabéticos, pero sí en el grupo de individuos sanos.
4. Sería necesario estudiar otras variables como podrían ser algunos factores externos: la ingesta de glucosa diaria, el índice glucémico o insulínico de los alimentos en la dieta, tipo de actividad física realizada. Así como parámetros metabólicos, por ejemplo la

hemoglobina glicosilada (HbA1c), un parámetro que mide el grado de glicosidación que sufre la molécula de hemoglobina durante la vida del hematíe (unos 180 días) y que por tanto, constituye un promedio de la concentración de glucosa en sangre que el individuo ha mantenido durante ese tiempo, reflejando mejor la glucemia que la medida puntual de una concentración de glucose en ayunas.

Otras cuestiones que nos podríamos haber planteado son:

1. ¿Existe relación entre los niveles de insulina en plasma y alguno de los parámetros de adiposidad?
2. Como hemos mencionado, sería necesario dividir la muestra entre grupos obesos-no obesos y analizar de nuevo las diferencias entre diabéticos o no diabéticos de algunos de los parámetros metabólicos(insulina, IR, leptina,…).

SECCIÓN 2 (2 PUNTOS)

Hemos realizado un aplicación con Shiny que muestra la tabla completa de datos, varios gráficos interactivos y además calcula el índice de resistencia a insulina si introducimos los datos necesarios para ello.

Shiny app Los datos y los archivos estan disponibles en: https://github.com/gititub/Shiny-app/tree/master/InsulinApp

La aplicación consta de un archivo server y un archivo UI. En el archivo server se define la parte lógica, usando shinyserver para definir una función que relaciona una serie de inputs con outputs. El input es la tabla de datos, y los outputs son distintos tipos de gráficos creados con ggplot, así como el resultado del cálculo del índice de resistencia a insulina HOMA en función de dos inputs (glucosa e insulina).

En el archivo UI (interfaz de usuario) se define mediante shinyUI el aspecto de la aplicación. Se incluyen varios tabpanels con la tabla completa de datos, los gráficos interactivos con sus checkbox y unos sliders para seleccionar los valores para el cálculo del índice HOMA.