Informe de la Práctica de Probabilidades y Estadística II

Grupo nº47

24 de Mayo - 2021

Índice

Cuadro 1: Miembros de GII-4F1T

Nombre	Apellidos	Nº Matrícula	Email
Álvaro	Cabo Ciudad	200172	alvaro.cabo@alumnos.upm.es
Yoan	Crul	200377	y.crul@alumnos.upm.es
Pablo	Brasero Gonzalez	200045	pablo.braserogonzalez@alumnos.upm.es
Jorge	García Plaza	200002	jorge.gplaza@alumnos.upm.es

Introducción y uso de librerías

Estas son las librerías que vamos a utilizar en el proyecto:

```
require(readr)
require(e1071)
require(MASS)

#extra
require(ggplot2)
require(fitdistrplus)
require(dplyr)
require(stests)
```

La lista incluye algunas de librerías recomendadas

- Readr para leer el csv, e1071 para summary y funciones estadísticas entre otras

Además de otras que hemos ido necesitando a medida que el proyecto iba creciendo

- Librerías gráficas o mejora de sintaxis: ggplot2, tidyverse
- Fitdistrplus, que a nuestro juicio hacía un mejor trabajo de ajuste que MASS
- stests, que utilizamos para calcular el intervalo de confianza de las varianzas y razón de varianzas

1. Identificacion de Modelo y Muestreo

1.a. Ajuste del modelo

El primer paso era importar el dataset, usando read_csv

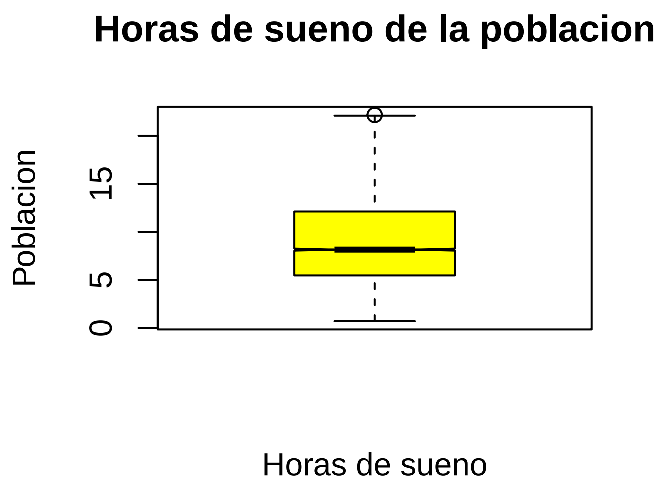
```
#En mi caso el dataset se encuentra en la carpeta de trabajo
Data <- read_csv(paste(getwd(),"PYE2DataSet47.csv", sep='/'))
```

Para proceder a una primera descripción de los datos, empezando por *Sleeptime*

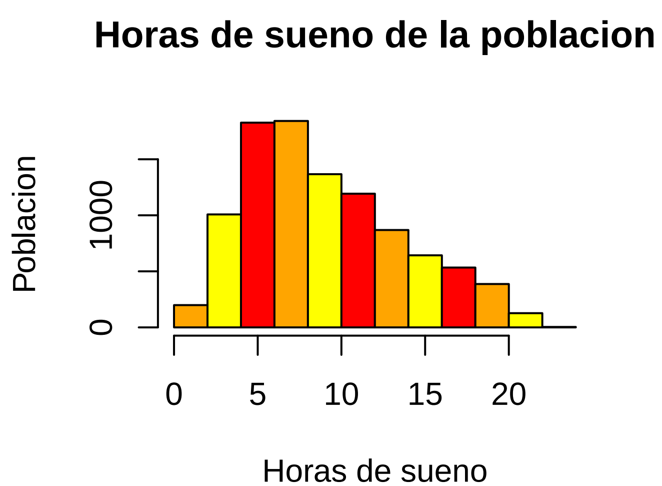
```
print(summary(Data$sleeptime))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.7064  5.4652  8.1546  9.1007 12.1171 22.1617
```

```r
boxplot(Data$sleeptime,
              xlab="Horas de sueno", ylab="Poblacion", main="Horas de sueno de la poblacion",
              notch = TRUE, col="yellow")
```




```r
hist(Data$sleeptime,
          xlab="Horas de sueno", ylab="Poblacion", main="Horas de sueno de la poblacion",
          col=c("orange", "yellow", "red"))
```

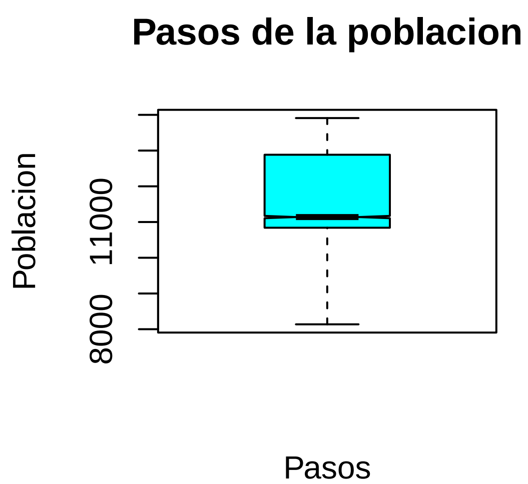


Y ahora *Number of Steps*

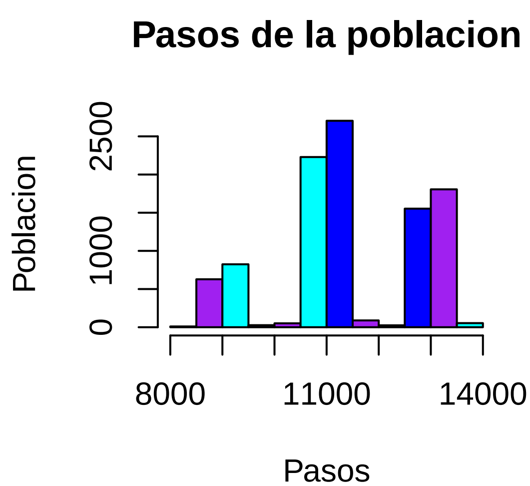
```r
print(summary(Data$sleeptime))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.7064  5.4652  8.1546  9.1007 12.1171 22.1617
```

```
boxplot(Data$steps,
             xlab="Pasos", ylab="Poblacion", main="Pasos de la poblacion",
             notch = TRUE, col="cyan")
```




```
hist(Data$steps,
          xlab="Pasos", ylab="Poblacion", main="Pasos de la poblacion",
          col=c("blue", "purple", "cyan"))
```

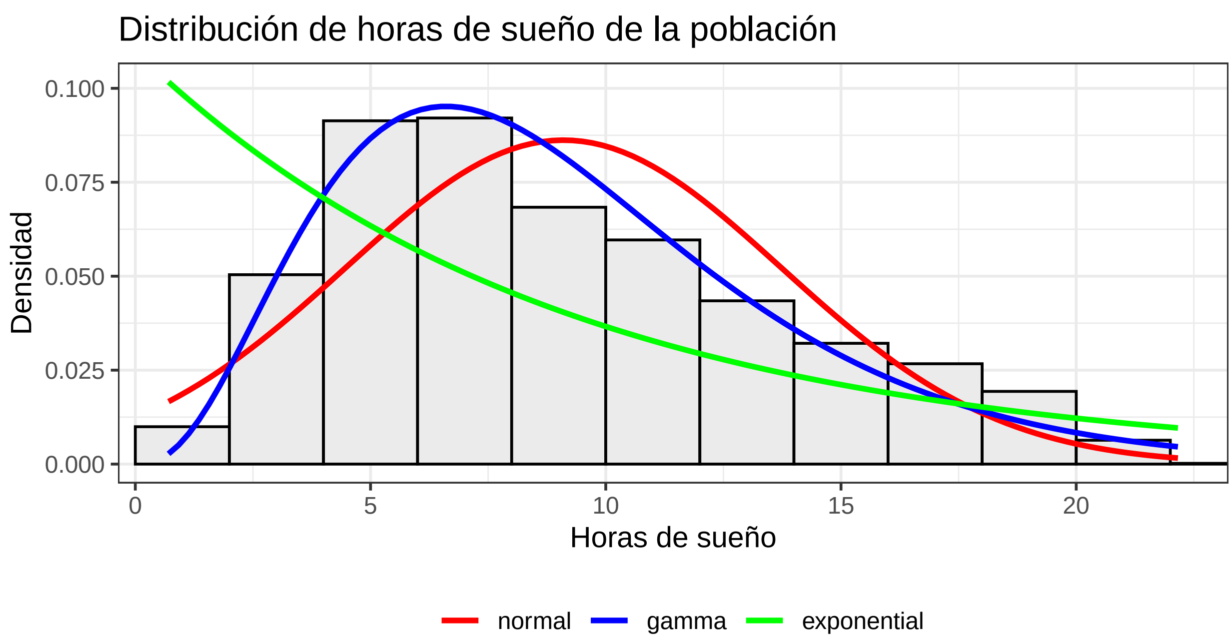


1.1. Ajuste de Sleeptime Usando la librería fitdistrplus ajustamos los datos a las 3 distribuciones pedidas y observamos sus resultados

Ploteado de distribuciones

Para entender los datos, vamos a graficar un histograma y sobre él las 3 distribuciones a las que los hemos ajustado

```r
p <- denscomp(
        list(normal, gamma, exponential),
        legendtext = c("normal", "gamma", "exponential"),
        xlab = "Horas de sueño",
        ylab = "Densidad",
        fitcol = c("red", "blue", "green"),
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
        addlegend = FALSE)
      #Retocamos el plot para hacerlo más legible
      p <- p +
        ggplot2::ggtitle("Distribución de horas de sueño de la población") +
        theme_bw() +
        theme(legend.position = "bottom")
      p
```



Se ve claramente que la distribución que mejor modela nuestros datos es la gamma, y comprobamos el ajuste evaluando la hipótesis de normalidad utilizando el *Test de Kolmogorov-Smirnov*

```r
d <- as.vector(Data$sleeptime) #vector de 100 elementos del dataset
```

```r
test <- ks.test(d, pnorm, normal$estimate[1], normal$estimate[2])

  print(test)
```

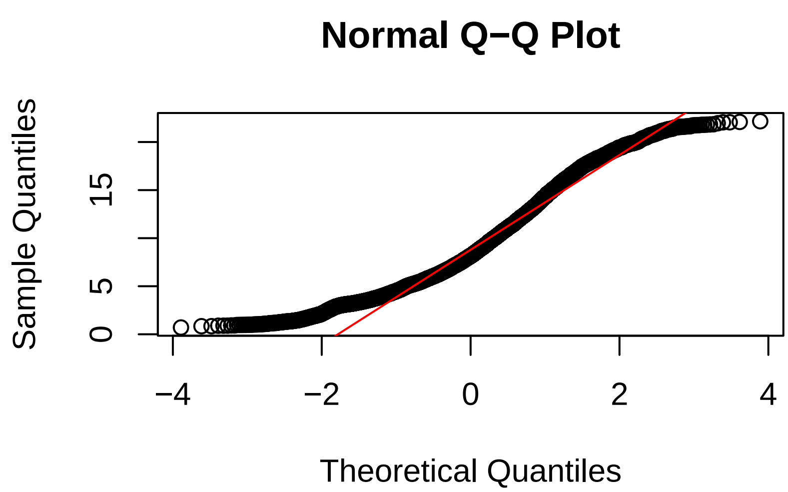
```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  d
```

```
## D = 0.083692, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

Al ser una muestra muy grande, era altamente probable que hubiera rechazo de hipótesis de normalidad

- Comparamos el resultado del test con lo obtenido al graficar los datos con un Q-Q plot

```
qqnorm(Data$sleeptime)
qqline(Data$sleeptime, col="red")
```



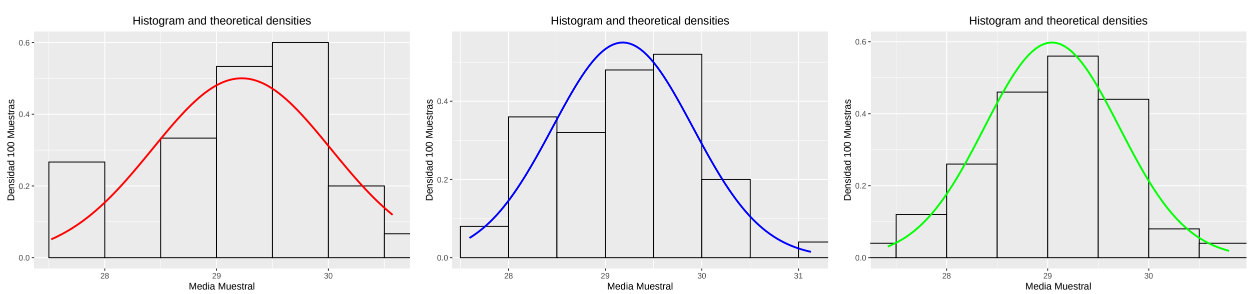
Vemos que no se ajusta a la linea de la normal, por lo que no va a haber hipótesis de normalidad

1.2. Muestreo de datos

Medias muestrales Gráficos de las Medias muestrales ajustadas a la normal

```
#Función de cáculo de medias muestrales de una muestra

mediaMuestral <-function(matrix){
  media <- c()
  for (i in 1:ncol(matrix)){
     media[i] <- mean(matrix[,i])}
  return(media)
}

m1 <-fitdist(mediaMuestral(muestras1), "norm")
m2 <-fitdist(mediaMuestral(muestras2), "norm")
m3 <-fitdist(mediaMuestral(muestras3), "norm")
p1 <- denscomp(
        m1,
        xlab = "Media Muestral",
        ylab = "Densidad 100 Muestras",
```

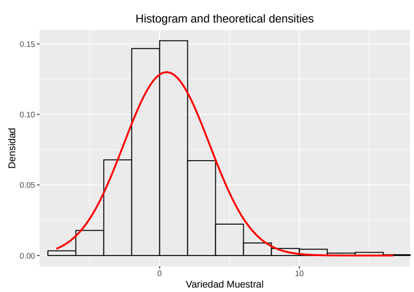
```
        fitcol = "red",
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
        addlegend = FALSE)

p2<- denscomp(
        m2,
        xlab =  "Media Muestral",
        ylab = "Densidad 100 Muestras",
        fitcol = "blue",
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
        addlegend = FALSE)

p3<-
   denscomp(
        m3,
        xlab =  "Media Muestral",
        ylab = "Densidad 100 Muestras",
        fitcol = "green",
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
        addlegend = FALSE)
p1
p2
p3
```

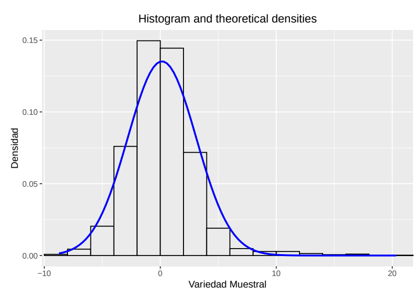


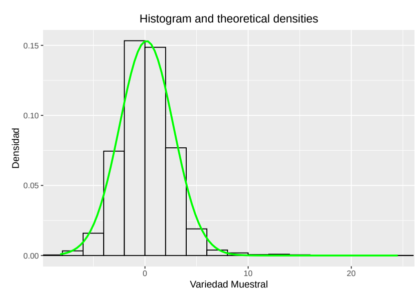
Varianzas muestrales Gráficos de las Varianzas muestrales ajustadas a la normal

```
m1v <-fitdist(as.vector(var(muestras1)), "norm")
m2v <- fitdist(as.vector(var(muestras2)), "norm")
p2v<- denscomp(
        m2v,
        xlab =  "Variedad Muestral",
        ylab = "Densidad",
        fitcol = "blue",
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
```

```
        addlegend = FALSE)


p1v <- denscomp(
        m1v,
        xlab = "Variedad Muestral",
        ylab = "Densidad",
        fitcol = c("red"),
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
        addlegend = FALSE)
m3v <- fitdist(as.vector(var(muestras3)), "norm")
p3v<-
   denscomp(
        m3v,
        xlab =  "Variedad Muestral",
        ylab = "Densidad",
        fitcol = "green",
        fitlty = 1,
        xlegend = "topright",
        plotstyle = "ggplot",
        addlegend = FALSE)

p1v
p2v
p3v
```



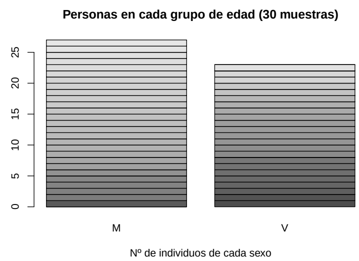


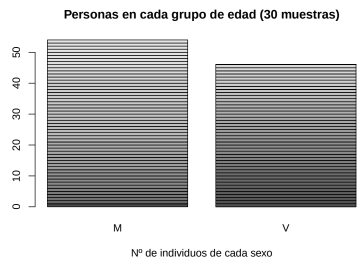


Proporción muestral entre Hombres y mujeres Para este caso he cogido 2 conjuntos de los 3 tamaños exigidos (30, 50, 100) y los he comparado

```
#Histograma hombre


barplot(muestras1,
     xlab="Nº de individuos de cada sexo",
     main="Personas en cada grupo de edad (30 muestras)",
)

barplot(muestras2,
     xlab="Nº de individuos de cada sexo",
     main="Personas en cada grupo de edad (30 muestras)",
```

```
)

barplot(muestras3,
     xlab="Nº de individuos de cada sexo",
     main="Personas en cada grupo de edad (30 muestras)",
)
```





2. Estimación Clásica (puntual, intervalos)

2.a. Estimación Puntual

Calculamos la media y varianza de Sleep y Steps de todos los datos

```
set.seed(2021)
mediaSleepAll <- mean(Data$sleeptime)
cat("Media Sleep: ", mediaSleepAll)
```

```
## Media Sleep:  9.100708
```

```
cat("Varianza Sleep: ", varSleepAll <- var(Data$sleeptime))
```

```
## Varianza Sleep:  21.41773
```

```
cat("Media Steps: ", mediaStepsAll <- mean(Data$steps))
```

```
## Media Steps:  11418.02
```

```
cat("Varianza Steps: ", varStepsAll <- var(Data$steps))
```

```
## Varianza Steps:  1860127
```

Calculamos la media y varianza de Sleep y Steps de una muestra de 20

```
set.seed(2021)
size <- 20
muestraSleep <- sample(Data$sleeptime, size=size)
muestraSteps <- sample(Data$steps, size=size)
cat("Media Sleep (20): ", mediaSleep20 <- mean(muestraSleep))
```

```
## Media Sleep (20):  9.055677
```

```
cat("Varianza Sleep (20): ", varSleep20 <- var(muestraSleep))
```

```
## Varianza Sleep (20):  18.0196
```

```
cat("Media Steps (20): ", mediaSteps20 <- mean(muestraSteps))
```

```
## Media Steps (20):  11669.56
```

```
cat("Varianza Steps (20): ", varSteps20 <- var(muestraSteps))
```

```
## Varianza Steps (20):  2038054
```

Calculamos la media y varianza de Sleep y Steps para todas las mujeres

```r
mujeres <- Data[Data$Sex == "M", ]
cat("Media Sleep (Mujeres): ", mediaSleepMujeres <- mean(mujeres$sleeptime))
```

```
## Media Sleep (Mujeres):  10.09315
```

```r
cat("Varianza Sleep (Mujeres): ", varSleepMujeres <- var(mujeres$sleeptime))
```

```
## Varianza Sleep (Mujeres):  20.43212
```

```r
cat("Media Steps (Mujeres): ", mediaStepsMujeres <- mean(mujeres$steps))
```

```
## Media Steps (Mujeres):  12407.23
```

```r
cat("Varianza Steps (Mujeres): ", varStepsMujeres <- var(mujeres$steps))
```

```
## Varianza Steps (Mujeres):  905847.3
```

Calculamos la media y la varianza para una muestra de 20 mujeres

```r
set.seed(2021)
size <- 20
muestraSleepMujeres20 <- sample(mujeres$sleeptime, size=size)
muestraStepsMujeres20 <- sample(mujeres$steps, size=size)
cat("Media Sleep (Mujeres, 20): ", mediaSleepMujeres20 <- mean(muestraSleep))
```

```
## Media Sleep (Mujeres, 20):  9.055677
```

```r
cat("Varianza Sleep (Mujeres, 20): ", varSleepMujeres20 <- var(muestraSleep))
```

```
## Varianza Sleep (Mujeres, 20):  18.0196
```

```r
cat("Media Steps (Mujeres, 20): ", mediaStepsMujeres20 <- mean(muestraSteps))
```

```
## Media Steps (Mujeres, 20):  11669.56
```

```r
cat("Varianza Steps (Mujeres, 20): ", varStepsMujeres20 <- var(muestraSteps))
```

```
## Varianza Steps (Mujeres, 20):  2038054
```

Calculamos la media y varianza de Sleep y Steps para todos los hombres

```r
hombres <- Data[Data$Sex =="V", ]
cat("Media Sleep (Hombres): ", mediaSleepHombres <- mean(hombres$sleeptime))
```

```
## Media Sleep (Hombres):  8.120108
```

```r
cat("Varianza Sleep (Hombres): ", varSleepHombres <- var(hombres$sleeptime))
```

```
## Varianza Sleep (Hombres):  20.4607
```

```r
cat("Media Steps (Hombres): ", mediaStepsHombres <- mean(hombres$steps))
```

```
## Media Steps (Hombres):  10440.6
```

```r
cat("Varianza Steps (Hombres): ", varStepsHombres <- var(hombres$steps))
```

```
## Varianza Steps (Hombres):  880795.1
```

Calculamos la media y la varianza de Sleep y Steps para una muestra de 20 hombres

```r
set.seed(2021)
size <- 20
muestraSleepHombres20 <- sample(hombres$sleeptime, size=size)
muestraStepsHombres20 <- sample(hombres$steps, size=size)
cat("Media Sleep (Hombres, 20): ", mediaSleepHombres20 <- mean(muestraSleep))
```

```
## Media Sleep (Hombres, 20):  9.055677
```

```r
cat("Varianza Sleep (Hombres, 20): ", varSleepHombres20 <- var(muestraSleep))
```

```
## Varianza Sleep (Hombres, 20):  18.0196
```

```r
cat("Media Steps (Hombres, 20): ", mediaStepsHombres20 <- mean(muestraSteps))
```

```
## Media Steps (Hombres, 20):  11669.56
```

```r
cat("Varianza Steps (Hombres, 20): ", varStepsHombres20 <- var(muestraSteps))
```

```
## Varianza Steps (Hombres, 20):  2038054
```

2.b Estimación de intervalos

1) Estimacion del intervalo de confianza para la media, varianza, proporcion, al nivel de confianza 90%,95% y 99 % Calculamos los intervalos de confianza de la media de sleep y steps de una muestra de 20 Hombres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```r
cat("Intervalo de confianza del 90% de la media Sleep de una muestra de 20 Hombres:
    ", intervalo90MediaSleepHombres20 <- t.test(x=muestraSleepHombres20, conf.level = 0.90)$conf.int)
```

```
## Intervalo de confianza del 90% de la media Sleep de una muestra de 20 Hombres:
##      6.556542 10.35847
```

```r
cat("Intervalo de confianza del 95% de la media Sleep de una muestra de 20 Hombres:
    ", intervalo95MediaSleepHombres20 <- t.test(x=muestraSleepHombres20, conf.level = 0.95)$conf.int)
```

```
## Intervalo de confianza del 95% de la media Sleep de una muestra de 20 Hombres:
##      6.15649 10.75852
```

```r
cat("Intervalo de confianza del 99% de la media Sleep de una muestra de 20 Hombres:
    ", intervalo99MediaSleepHombres20 <- t.test(x=muestraSleepHombres20, conf.level = 0.99)$conf.int)
```

```
## Intervalo de confianza del 99% de la media Sleep de una muestra de 20 Hombres:
##      5.312268 11.60275
```

```r
cat("Intervalo de confianza del 90% de la media Steps de una muestra de 20 Hombres:
    ", intervalo90MediaStepsHombres20 <- t.test(x=muestraStepsHombres20, conf.level = 0.90)$conf.int)
```

```
## Intervalo de confianza del 90% de la media Steps de una muestra de 20 Hombres:
##      9804.827 10607.92
```

```r
cat("Intervalo de confianza del 95% de la media Steps de una muestra de 20 Hombres:
    ", intervalo95MediaStepsHombres20 <- t.test(x=muestraStepsHombres20, conf.level = 0.95)$conf.int)
```

```
## Intervalo de confianza del 95% de la media Steps de una muestra de 20 Hombres:
##      9720.323 10692.42
```

```r
cat("Intervalo de confianza del 99% de la media Steps de una muestra de 20 Hombres:
    ", intervalo99MediaStepsHombres20 <- t.test(x=muestraStepsHombres20, conf.level = 0.99)$conf.int)
```

```
## Intervalo de confianza del 99% de la media Steps de una muestra de 20 Hombres:
##      9541.997 10870.75
```

Calculamos los intervalos de confianza de la varianza de sleep y steps de una muestra de 20 Hombres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```r
cat("Intervalo de confianza del 90% de la varianza Sleep de una muestra de 20 Hombres:
    ", intervalo90VarSleepHombres20 <- var.test(x=muestraSleepHombres20, muestraSleepHombres20,conf.leve
```

```
## Intervalo de confianza del 90% de la varianza Sleep de una muestra de 20 Hombres:
##      0.4612011 2.168252
```

```r
cat("Intervalo de confianza del 95% de la varianza Sleep de una muestra de 20 Hombres:
    ", intervalo95VarleepHombres20 <- var.test(x=muestraSleepHombres20, muestraSleepHombres20,conf.level
```

```
## Intervalo de confianza del 95% de la varianza Sleep de una muestra de 20 Hombres:
##      0.3958122 2.526451
```

```r
cat("Intervalo de confianza del 99% de la varianza Sleep de una muestra de 20 Hombres:
    ", intervalo99VarSleepHombres20 <- var.test(x=muestraSleepHombres20, muestraSleepHombres20,conf.leve
```

```
## Intervalo de confianza del 99% de la varianza Sleep de una muestra de 20 Hombres:
##      0.2913965 3.43175
```

```r
cat("Intervalo de confianza del 90% de la varianza Steps de una muestra de 20 Hombres:
    ", intervalo90VarStepsHombres20 <- var.test(x=muestraStepsHombres20,muestraSleepHombres20, conf.leve
```

```
## Intervalo de confianza del 90% de la varianza Steps de una muestra de 20 Hombres:
##      20578.37 96745.39
```

```r
cat("Intervalo de confianza del 95% de la varianza Steps de una muestra de 20 Hombres:
    ", intervalo95VarStepsHombres20 <- var.test(x=muestraStepsHombres20,muestraSleepHombres20, conf.leve
```

```
## Intervalo de confianza del 95% de la varianza Steps de una muestra de 20 Hombres:
##      17660.77 112727.9
```

```r
cat("Intervalo de confianza del 99% de la varianza Steps de una muestra de 20 Hombres:
    ", intervalo99VarStepsHombres20 <- var.test(x=muestraStepsHombres20,muestraSleepHombres20, conf.leve
```

```
## Intervalo de confianza del 99% de la varianza Steps de una muestra de 20 Hombres:
##      13001.84 153121.5
```

Calculamos los intervalos de confianza de la media de sleep y steps de una muestra de 20 Mujeres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```r
cat("Intervalo de confianza del 90% de la media Sleep de una muestra de 20 Mujeres:
    ", intervalo90MediaSleepMujeres20 <- t.test(x=muestraSleepMujeres20, conf.level = 0.90)$conf.int)
```

```
## Intervalo de confianza del 90% de la media Sleep de una muestra de 20 Mujeres:
##      7.967601 11.40785
```

```r
cat("Intervalo de confianza del 95% de la media Sleep de una muestra de 20 Mujeres:
    ", intervalo95MediaSleepMujeres20 <- t.test(x=muestraSleepMujeres20, conf.level = 0.95)$conf.int)
```

```
## Intervalo de confianza del 95% de la media Sleep de una muestra de 20 Mujeres:
##      7.605605 11.76985
```

```r
cat("Intervalo de confianza del 99% de la media Sleep de una muestra de 20 Mujeres:
    ", intervalo99MediaSleepMujeres20 <- t.test(x=muestraSleepMujeres20, conf.level = 0.99)$conf.int)
```

```
## Intervalo de confianza del 99% de la media Sleep de una muestra de 20 Mujeres:
##      6.841694 12.53376
```

```r
cat("Intervalo de confianza del 90% de la media Steps de una muestra de 20 Mujeres:
    ", intervalo90MediaStepsMujeres20 <- t.test(x=muestraStepsMujeres20, conf.level = 0.90)$conf.int)
```

```
## Intervalo de confianza del 90% de la media Steps de una muestra de 20 Mujeres:
##      11547.69 12387.73
```

```r
cat("Intervalo de confianza del 95% de la media Steps de una muestra de 20 Mujeres:
    ", intervalo95MediaStepsMujeres20 <- t.test(x=muestraStepsMujeres20, conf.level = 0.95)$conf.int)
```

```
## Intervalo de confianza del 95% de la media Steps de una muestra de 20 Mujeres:
##      11459.29 12476.12
```

```r
cat("Intervalo de confianza del 99% de la media Steps de una muestra de 20 Mujeres:
    ", intervalo99MediaStepsMujeres20 <- t.test(x=muestraStepsMujeres20, conf.level = 0.99)$conf.int)
```

```
## Intervalo de confianza del 99% de la media Steps de una muestra de 20 Mujeres:
##      11272.76 12662.65
```

Calculamos los intervalos de confianza de la varianza de sleep y steps de una muestra de Mujeres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```r
cat("Intervalo de confianza del 90% de la varianza Sleep de una muestra de 20 Mujeres:
    ", intervalo90VarSleepMujeres20 <- var.test(x=muestraSleepMujeres20,muestraSleepMujeres20, conf.leve
```

```
## Intervalo de confianza del 90% de la varianza Sleep de una muestra de 20 Mujeres:
##      0.4612011 2.168252
```

```r
cat("Intervalo de confianza del 95% de la varianza Sleep de una muestra de 20 Mujeres:
    ", intervalo95VarSleepMujeres20 <- var.test(x=muestraSleepMujeres20,muestraSleepMujeres20, conf.leve
```

```
## Intervalo de confianza del 95% de la varianza Sleep de una muestra de 20 Mujeres:
##      0.3958122 2.526451
```

```r
cat("Intervalo de confianza del 99% de la varianza Sleep de una muestra de 20 Mujeres:
    ", intervalo99VarSleepMujeres20 <- var.test(x=muestraSleepMujeres20,muestraSleepMujeres20, conf.leve
```

```
## Intervalo de confianza del 99% de la varianza Sleep de una muestra de 20 Mujeres:
##      0.2913965 3.43175
```

```r
cat("Intervalo de confianza del 90% de la varianza Steps de una muestra de 20 Mujeres:
    ", intervalo90VarStepsMujeres20 <- var.test(x=muestraSleepMujeres20,muestraSleepMujeres20, conf.leve
```

```
## Intervalo de confianza del 90% de la varianza Steps de una muestra de 20 Mujeres:
##      0.4612011 2.168252
```

```r
cat("Intervalo de confianza del 95% de la varianza Steps de una muestra de 20 Mujeres:
    ", intervalo95VarStepsMujeres20 <- var.test(x=muestraSleepMujeres20,muestraSleepMujeres20, conf.leve
```

```
## Intervalo de confianza del 95% de la varianza Steps de una muestra de 20 Mujeres:
##      0.3958122 2.526451
```

```r
cat("Intervalo de confianza del 99% de la varianza Steps de una muestra de 20 Mujeres:
    ", intervalo99VarStepsMujeres20 <- var.test(x=muestraSleepMujeres20,muestraSleepMujeres20, conf.leve
```

```
## Intervalo de confianza del 99% de la varianza Steps de una muestra de 20 Mujeres:
##      0.2913965 3.43175
```

2) Estimacion del intervalo de confianza para la diferencia de medias, razon de varianzas, proporcion, al nivel de confianza 90%, 95% y 99% Calculamos los intervalos de confianza de la diferencia de la media de sleep de una muestra de 20 hombres y una muestra de 20 muejres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```r
cat("Intervalo de confianza del 90% de la diferencia de las
medias sleep entre hombres y mujeres:
    ", intervalo90DifMediasHombres20 <- t.test(x=muestraStepsMujeres20, y=muestraStepsHombres20, paired
```

```
## Intervalo de confianza del 90% de la diferencia de las
## medias sleep entre hombres y mujeres:
##      1194.734 2327.933
```

```r
cat("Intervalo de confianza del 95% de la diferencia de las
medias sleep entre hombres y mujeres:
    ", intervalo95DifMediasHombres20 <- t.test(x=muestraStepsMujeres20, y=muestraStepsHombres20, paired
```

```
## Intervalo de confianza del 95% de la diferencia de las
## medias sleep entre hombres y mujeres:
##      1080.983 2441.683
```

```r
cat("Intervalo de confianza del 99% de la diferencia de las
medias sleep entre hombres y mujeres:
    ", intervalo99DifMediasHombres20 <- t.test(x=muestraStepsMujeres20, y=muestraStepsHombres20, paired
```

```
## Intervalo de confianza del 99% de la diferencia de las
## medias sleep entre hombres y mujeres:
##      850.0083 2672.659
```

Calculamos los intervalos de confianza de la diferencia de la media de steps de una muestra de 20 hombres y una muestra de 20 muejres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```r
cat("Intervalo de confianza del 90% de la diferencia de las
medias sleep entre hombres y mujeres:
    ", intervalo90DifMediasHombres20 <- t.test(x=muestraStepsMujeres20, y=muestraStepsHombres20, paired
```

```
## Intervalo de confianza del 90% de la diferencia de las
## medias sleep entre hombres y mujeres:
##      1194.734 2327.933
```

```r
cat("Intervalo de confianza del 95% de la diferencia de las
medias sleep entre hombres y mujeres:
    ", intervalo95DifMediasHombres20 <- t.test(x=muestraStepsMujeres20, y=muestraStepsHombres20, paired
```

```
## Intervalo de confianza del 95% de la diferencia de las
## medias sleep entre hombres y mujeres:
##      1080.983 2441.683
```

```r
cat("Intervalo de confianza del 99% de la diferencia de las
medias sleep entre hombres y mujeres:
    ", intervalo99DifMediasHombres20 <- t.test(x=muestraStepsMujeres20, y=muestraStepsHombres20, paired
```

```
## Intervalo de confianza del 99% de la diferencia de las
## medias sleep entre hombres y mujeres:
##      850.0083 2672.659
```

Calculamos los intervalos de confianza de la razón de varianzas de sleep de una muestra de 20 hombres y una muestra de 20 mujeres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```
cat("Intervalo de confianza del 90% de la razón de varianzas de Sleep entre hombres y mujeres:
    ", intervalo90RazVarSleep <- var.test(x=muestraStepsHombres20, y=muestraStepsMujeres20, conf.level=0
```

```
## Intervalo de confianza del 90% de la razón de varianzas de Sleep entre hombres y mujeres:
##      0.4215206 1.981701
```

```
cat("Intervalo de confianza del 95% de la razón de varianzas de Sleep entre hombres y mujeres:
    ", intervalo90RazVarSleep <- var.test(x=muestraStepsHombres20, y=muestraStepsMujeres20, conf.level=0
```

```
## Intervalo de confianza del 95% de la razón de varianzas de Sleep entre hombres y mujeres:
##      0.3617575 2.309082
```

```
cat("Intervalo de confianza del 99% de la razón de varianzas de Sleep entre hombres y mujeres:
    ", intervalo90RazVarSleep <- var.test(x=muestraStepsHombres20, y=muestraStepsMujeres20, conf.level=0
```

```
## Intervalo de confianza del 99% de la razón de varianzas de Sleep entre hombres y mujeres:
##      0.2663255 3.136492
```

Calculamos los intervalos de confianza de la razón de varianzas de steps de una muestra de 20 hombres y una muestra de 20 mujeres, a niveles de confianza 90%, 95%, 99%, suponiendo normalidad

```
cat("Intervalo de confianza del 90% de la razón de varianzas de Steps entre hombres y mujeres:
    ", intervalo90RazVarSleep <- var.test(x=muestraStepsHombres20, y=muestraStepsMujeres20, conf.level=0
```

```
## Intervalo de confianza del 90% de la razón de varianzas de Steps entre hombres y mujeres:
##      0.4215206 1.981701
```

```
cat("Intervalo de confianza del 95% de la razón de varianzas de Steps entre hombres y mujeres:
    ", intervalo90RazVarSleep <- var.test(x=muestraStepsHombres20, y=muestraStepsMujeres20, conf.level=0
```

```
## Intervalo de confianza del 95% de la razón de varianzas de Steps entre hombres y mujeres:
##      0.3617575 2.309082
```

```
cat("Intervalo de confianza del 99% de la razón de varianzas de Steps entre hombres y mujeres:
    ", intervalo90RazVarSleep <- var.test(x=muestraStepsHombres20, y=muestraStepsMujeres20, conf.level=0
```

```
## Intervalo de confianza del 99% de la razón de varianzas de Steps entre hombres y mujeres:
##      0.2663255 3.136492
```

3.- ESTIMACIÓN BAYESIANA

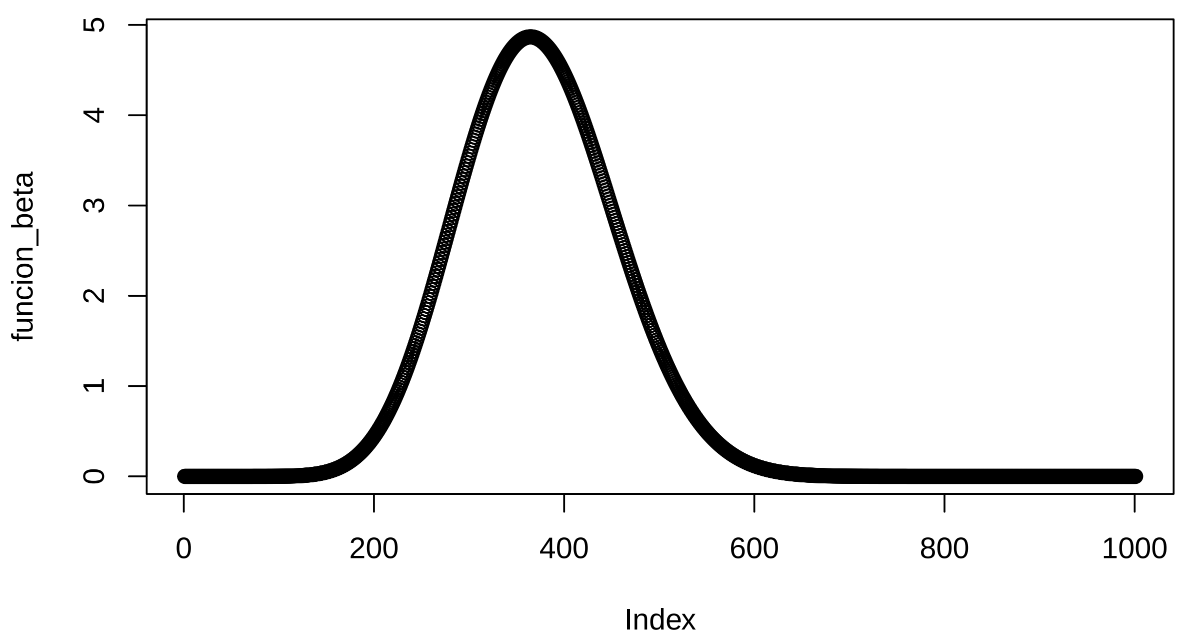
Vamos a crear ima muestra de tamaño 20 tal y como se necesita segun el enunciado

```
## [1] 16
```

```
## [1] 0.2
```

a) Obtener la pe tras la informacion aportada por la muestra

```
dummyVec_Esp <- ifelse(muestra20$Nation == "SP", 1,0)
aP=13     #alfa a posteriori (de mi distribución Beta a posteriori)
bP=22     #beta a posteriori (de mi distribución Beta a posteriori)


#necesitamos un vector para pasarselo a la funcion beta, lo calculamos y aplicamos la funcion
vector_beta <- seq(0, 1, by = 0.001)
funcion_beta <- dbeta(vector_beta, shape1 = aP, shape2 = bP)
plot(funcion_beta)
```



b) Obtener el IC con 95% de confianza para la pe

Para hallar el intervalo de confianza deseado usamos qbeta, siendo el primer parametro el valor del cuantil deseado, el segundo parametro es alfa y el tercer parametro el beta

```r
aP=13
bP=22
qbeta(0.975, aP, bP)
```

```
## [1] 0.5351136
```

```r
qbeta(0.025, aP, bP)
```

```
## [1] 0.2216654
```

c) Estimar con la muestra anterior, la estatura con varianza conocida.

```r
#Filtramos y vemos si df ha sufrido algun cambio
muestra20_esp_it_fr <- muestra20[muestra20$Nation=="SP" | muestra20$Nation=="IT" | muestra20$Nation=="F
head(muestra20_esp_it_fr)
```

```
## # A tibble: 6 x 10
##    ...1 name   Sex   Nation sleeptime  steps height weight   IMC   Age
##   <dbl> <chr>  <chr> <chr>      <dbl>  <dbl>  <dbl>  <dbl> <dbl> <dbl>
## 1  2045 P_4204 M     SP          3.48 10905.   161.   65.6  25.2  33.1
## 2  8592 U_6700 M     IT         16.7  13076.   175.   72.9  23.8  36.1
## 3  1624 J_5059 M     SP         13.3  11170.   173.   72.1  24.1  27.7
## 4  1553 E_5341 V     IT          3.54 10911.   166.   70.9  25.6  27.2
## 5  9573 C_100  M     IT          9.63 13028.   157.   63.5  25.6  28.6
## 6  6292 U_8719 V     IT          5.38 10981.   179.   77.7  24.3  28.7
```

```r
vec_alturas <- muestra20_esp_it_fr$height
mean(vec_alturas)
```

```
## [1] 168.1638
```

```r
var(vec_alturas)
```

```
## [1] 75.92628
```

```r
#Obtenemos cuartil 0.025 sabiendo que n = 14 pues el valor de df ha cambiado previamente.
qt(p=0.975, df = 3, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 3.182446
```

4.- CONTRASTES (PARAMÉTRICOS Y NO PARAMÉTRICOS)

4.1- CONTRASTES PARAMÉTRICOS

Contrastar si la media mu1 de Sample1 es Q1 <= mu1, con varianza desconocida (Q1 : cuartil 1 de la muestra)

```
set.seed(2022)
#Generamos las muestras aleatorias de tamaño 20
muestra1_apartado4 <- sample(Data$IMC,size=20)
muestra2_apartado4 <- sample(Data$IMC,size=20)
#Almacenamos el cuartil 1
q1 <- quantile(muestra1_apartado4,0.25)
#Siendo el cuartil 25% 24.60137, contrastamos
t.test(muestra1_apartado4, alternative="less", mu=q1)
```

```
##
##  One Sample t-test
##
## data:  muestra1_apartado4
## t = 2.6648, df = 19, p-value = 0.9923
## alternative hypothesis: true mean is less than 24.60137
## 95 percent confidence interval:
##      -Inf 25.21773
## sample estimates:
## mean of x
##  24.97518
```

```
#Puesto que p-value es 0.9923 y la media es 24.97518, no rechazamos la hipótesis nula
# y comprobamos que el cuartil 25% es menor que la media
```

Contrastar si la media mu1 de Sample1 es mu1 <= Q3, con varianza desconocida

```
q3 <- quantile(muestra1_apartado4, 0.75)
t.test(muestra1_apartado4, alternative="greater",mu=q3)
```

```
##
##  One Sample t-test
##
## data:  muestra1_apartado4
## t = -2.0206, df = 19, p-value = 0.9712
## alternative hypothesis: true mean is greater than 25.25862
## 95 percent confidence interval:
##  24.73262      Inf
## sample estimates:
## mean of x
##  24.97518
```

```
#Puesto que p-value es 0.9712, aceptamos la hipótesis nula de que la media es menor que el #tercer cuar
```

Contrastar si la varianza sigma2 de Sample1 es mayor que 1.0, con media desconocida

```
library(TeachingDemos)
sigma.test(muestra1_apartado4, sigma=1, alterantive="less")
```

```
##
##  One sample Chi-squared test for variance
##
## data:  muestra1_apartado4
## X-squared = 7.4773, df = 19, p-value = 0.0176
## alternative hypothesis: true variance is not equal to 1
## 95 percent confidence interval:
##  0.2276021 0.8395266
## sample estimates:
## var of muestra1_apartado4
##                 0.3935399
```

```
#aceptamos la alternativa de que es menor puesto que p-value es 0.0176 y #var(muestra1_apartado4) = 0.3
```

Contrastar si mu1 - mu2 = 0, con Sample1 y Sample2 respectivamente, con varianzas desconocidas

```
t.test(muestra1_apartado4,muestra2_apartado4, paired=TRUE)
```

```
##
##  Paired t-test
##
## data:  muestra1_apartado4 and muestra2_apartado4
## t = -0.1456, df = 19, p-value = 0.8858
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5142683  0.4473741
## sample estimates:
## mean of the differences
##             -0.03344711
```

```
#Como p-value = 0.8858, en un intervalo de confianza del 95%, aceptamos la hipostesis nula
#de que la diferencia de las medias es 0
```

Contrastar si varianza sigma1/ varianza sigma2 = 1, con Sample1 y Sample2 respectivamente

```
var.test(muestra1_apartado4,muestra2_apartado4)
```

```
##
##  F test to compare two variances
##
## data:  muestra1_apartado4 and muestra2_apartado4
## F = 0.81725, num df = 19, denom df = 19, p-value = 0.6645
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3234785 2.0647487
## sample estimates:
## ratio of variances
##          0.8172527
```

```
#Como p-value = 0.6645, en un intervalo de confianza del 95%, aceptamos la hipostesis nula
#de que la division de las varianzas es igual a 1
```

4.2- CONTRASTES PARAMÉTRICOS

Contrastar la normalidad de Sample, mediante el test de Pearson y el test de #Kolmogorov-Smirnov

```
set.seed(2022)
library(MASS)
#Creamos la muestra aleatoria de tamaño 20
muestra3_apartado4 <- sample(Data$IMC,size=20)
fitdistr(muestra3_apartado4,c("normal"))
```

```
##       mean           sd
##   24.97517790    0.61144326
##  ( 0.13672287) ( 0.09667767)
```

```
pnorm <- pnorm(20,24.97517790,0.61144326)
ks.test(muestra3_apartado4, pnorm, 24.97517790, 0.61144326)
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  muestra3_apartado4 and pnorm
## D = 1, p-value = 0.09524
## alternative hypothesis: two-sided
```

Contrastar la independencia de Sample, mediante el test de Durbin-Watson

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
muestra6_apartado4 <- sample(Data$IMC,size=20)
lm <- lm(IMC ~ Data$height + Data$weight, data = Data)
dwtest(lm)
```

```
##
##  Durbin-Watson test
##
## data:  lm
## DW = 1.9806, p-value = 0.1662
## alternative hypothesis: true autocorrelation is greater than 0
```

Contrastar la homogeneidad de Sample, mediante el test de Wilcoxon

```
muestra4_apartado4 <- sample(Data$IMC,size=20)
muestra5_apartado4 <- sample(Data$IMC,size=20)
wilcox.test(x = muestra4_apartado4, y = muestra5_apartado4,
            alternative = "two.sided", mu = 0,
            paired = FALSE, conf.int = 0.95)
```

```
##
##  Wilcoxon rank sum exact test
##
## data:  muestra4_apartado4 and muestra5_apartado4
## W = 200, p-value = 1
## alternative hypothesis: true location shift is not equal to 0
## 95 percent confidence interval:
##  -0.5152313  0.5490714
## sample estimates:
## difference in location
##           -0.002312251
```

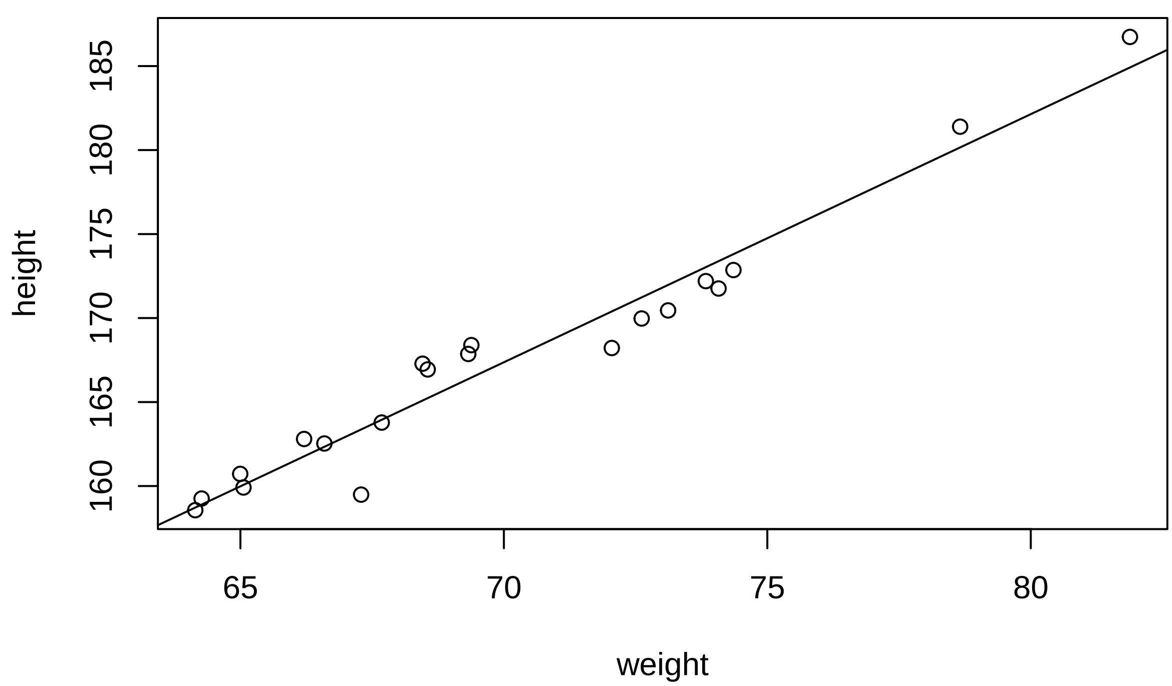
5.- REGRESIÓN LINEAL SIMPLE (ESTIMACIÓN Y CONTRASTE)

5.a.Estimación del modelo de regresión simple

```
set.seed(2022)
df <- data.frame(Data)

#  Muestra de n=20 del dataframe completo
df_20 <- df[sample(1:nrow(df),20),]

#Modelo de regresi?n lineal simple
regresion <- lm(height~weight, data=df_20)
summary(regresion)
```

```
##
## Call:
## lm(formula = height ~ weight, data = df_20)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8781 -1.0230  0.0209  1.2981  2.1921
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 63.98309    5.43957   11.76 6.96e-10 ***
## weight       1.47694    0.07739   19.08 2.16e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.637 on 18 degrees of freedom
## Multiple R-squared:  0.9529, Adjusted R-squared:  0.9503
## F-statistic: 364.2 on 1 and 18 DF,  p-value: 2.161e-13
```

```
plot(df_20$weight, df_20$height, xlab='weight', ylab='height')
abline(regresion)
```



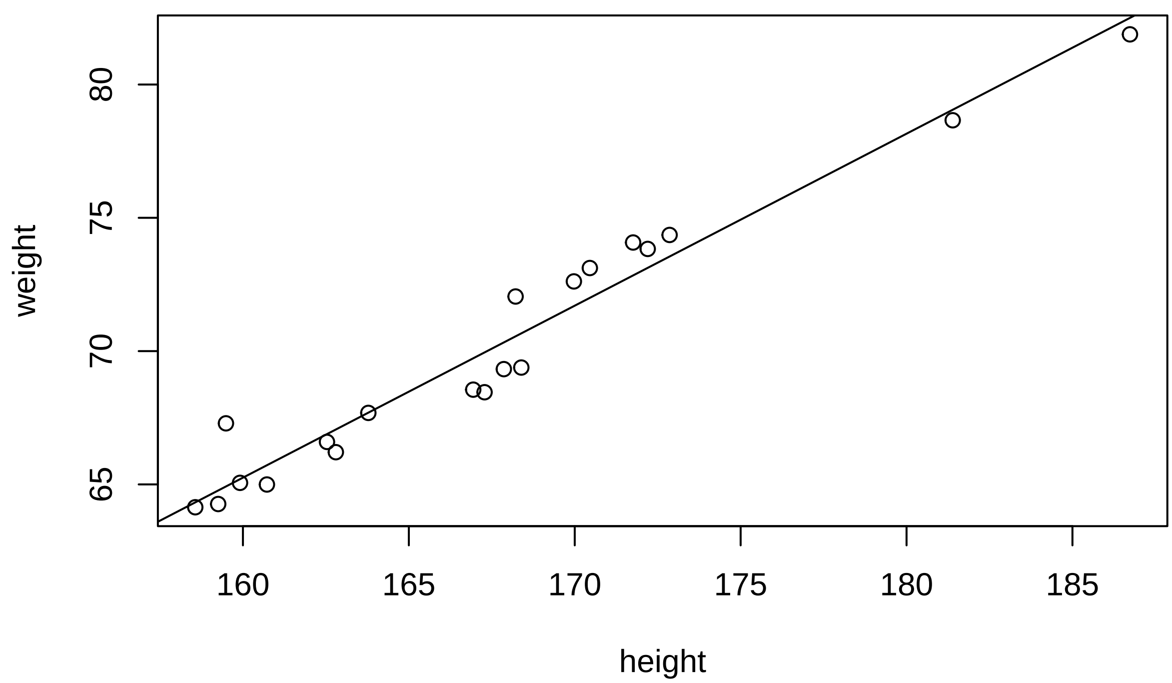

```
#Predicciones: Predecimos alturas para diferentes pesos
nuevos.pesos <- data.frame(weight =seq(60, 80))
predict(regresion, nuevos.pesos)
```

```
##        1        2        3        4        5        6        7        8
## 152.5992 154.0762 155.5531 157.0300 158.5070 159.9839 161.4609 162.9378
##        9       10       11       12       13       14       15       16
## 164.4147 165.8917 167.3686 168.8455 170.3225 171.7994 173.2763 174.7533
##       17       18       19       20       21
## 176.2302 177.7071 179.1841 180.6610 182.1380
```

```r
#Inferencia en el modelo de regresion simple
confint(regresion)
```

```
##                 2.5 %   97.5 %
## (Intercept) 52.554966 75.41121
## weight       1.314342  1.63953
```

```r
#Representamos intervalos de confianza para la respuesta media y los intervalos de predicci?n (en rojo)
plot(df_20$height, df_20$weight, xlab='height', ylab='weight')
abline(lm(weight~height, data=df_20))

ic <- predict(regresion, nuevos.pesos, interval = 'confidence')
lines(nuevos.pesos$weight, ic[,2], lty = 2)
lines(nuevos.pesos$weight, ic[,3], lty = 2)

ic <- predict(regresion, nuevos.pesos, interval = 'prediction')
lines(nuevos.pesos$weight, ic[,2], lty = 2, col= 'red')
lines(nuevos.pesos$weight, ic[,3], lty = 2, col = 'red')
```

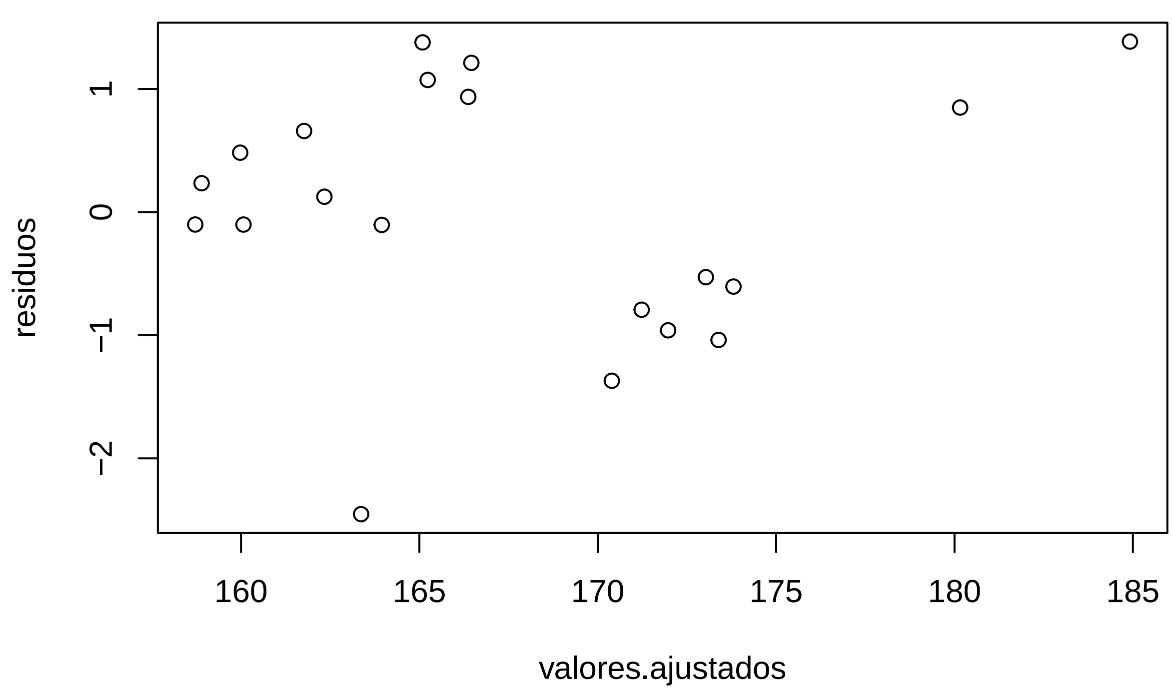


```r
# Tabla de an?lisis de la varianza de los errores con el comando ANOVA:
anova(regresion)
```

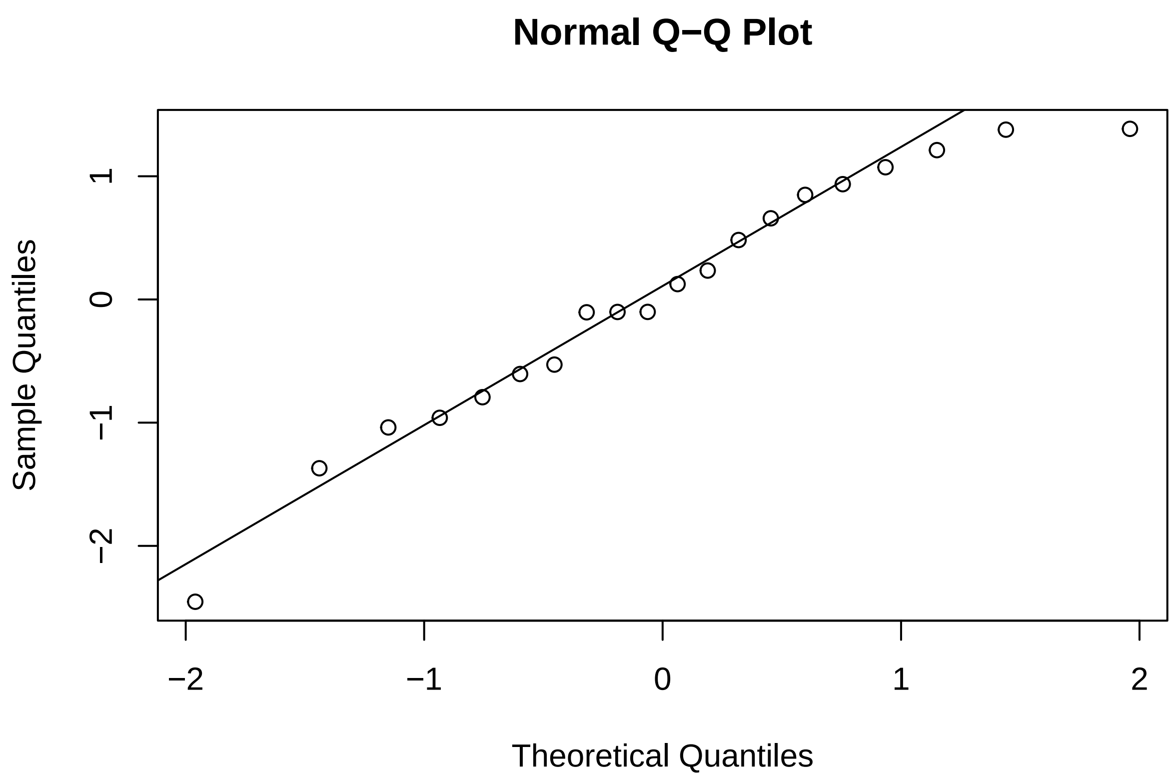
```
## Analysis of Variance Table
##
## Response: height
##           Df Sum Sq Mean Sq F value    Pr(>F)
## weight     1 976.31  976.31   364.2 2.161e-13 ***
## Residuals 18  48.25    2.68
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5.b.Contraste de regresión

```r
#Representamos los residuos estandarizados frente a los valores ajustados

residuos <- rstandard(regresion)
valores.ajustados <- fitted(regresion)
plot(valores.ajustados, residuos)
```




```r
#Comprobamos la hip?tesis de normalidad con un QQPlot
qqnorm(residuos)
qqline(residuos)
```


Normal Q−Q Plot
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2