01_Simulacion_Estadistica

Alicia Gil Matute

2024-09-16

1. EJERCICIOS SOBRE DISTRIBUCIONES

La wikipedia contiene excelente información sobre las funciones de densidad de probabilidad de las distribuciones uniforme, t, F y χ^2. En estas distribuciones aparecerán dos funciones, denotadas por Γ y β y llamadas funciones Gamma y Beta de Euler y que se definen de la siguiente manera:

$$\Gamma(x) = \int_0^{inf} t^{x-1}\ e^{-t}\, dt$$

$$Beta(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}$$

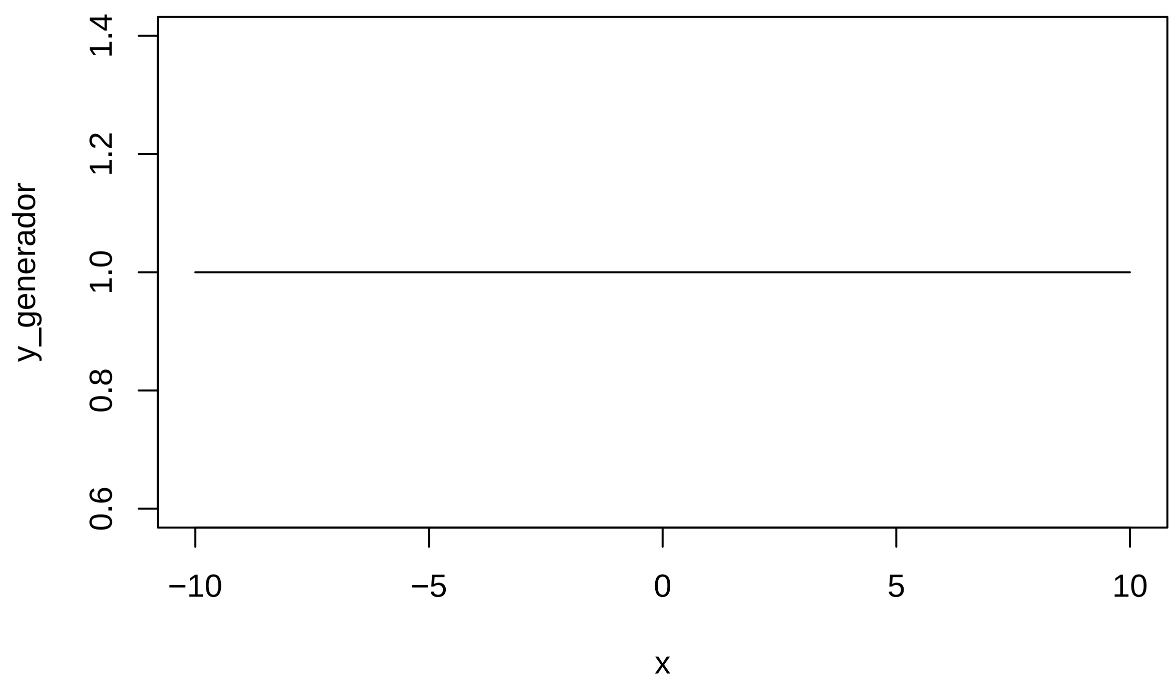
Pero ninguna de las dos es necesario implementarlas ya que están disponibles en R en las funciones gamma() y beta().

Siguiendo la lógica que he utilizado en el apartado anterior realiza los siguientes ejercicios: 1. Muestra que la función runif produce una distribución uniforme que coincide con la teórica 2. Muestra que rt genera distribuciones t que coinciden con la teórica 3. Muestra que rf genera distribuciones F que coinciden con la teórica 4. Muestra que rchisq genera distribuciones chisq que coinciden con la teórica Para mostrarlo, simplemente dibuja las distribuciones y comprueba que el histograma se aproxima a su distribución teórica. Para resolver bien los ejercicios recuerda que tienes que ajustar los valores máximos y mínimos de x para que representen la gráfica. Te será de ayuda conocer la moda (el valor más alto) de cada una de ellas:

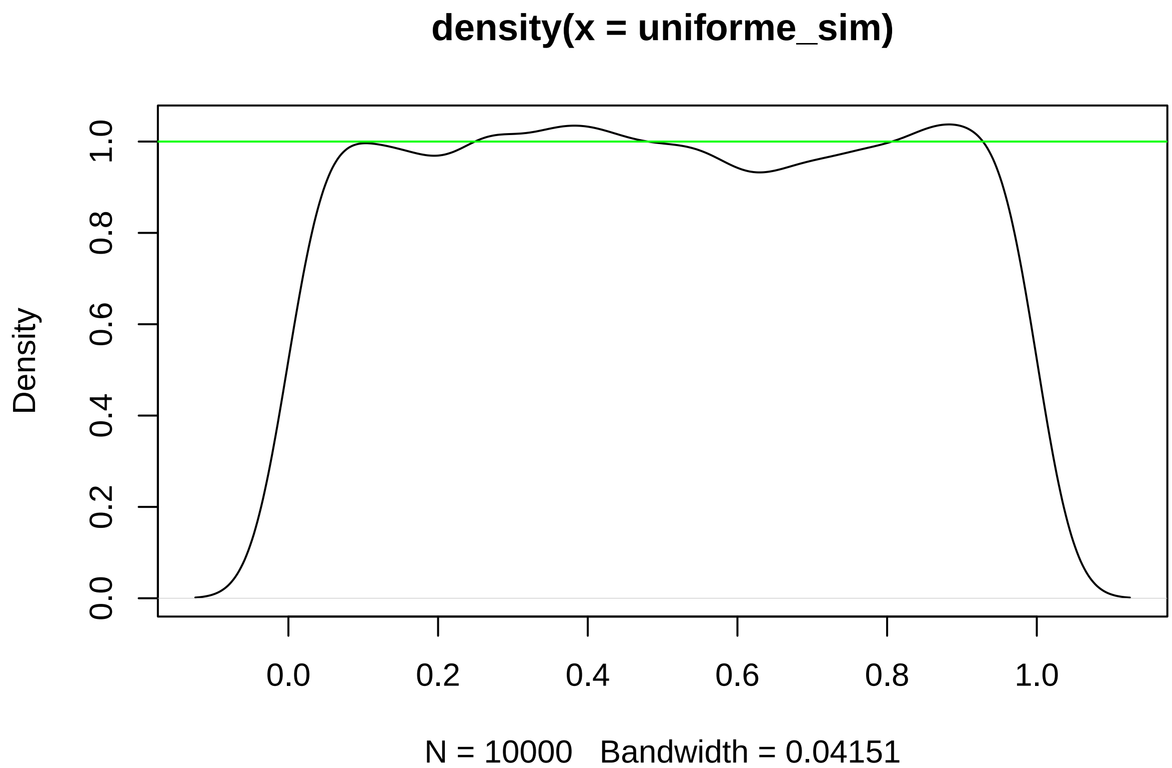
$$\text{Moda(t)} = 0$$
$$\text{Moda}(F_{gl1,gl2}) = \frac{gl_2}{gl_2 - 2} \text{ para } gl_2 > 2$$
$$\text{Moda}(\chi^2) = gl - 2 \text{ para } gl > 2$$

```r
#### 1- DISTRIBUCION UNIFORME

set.seed(1)

#teorica

uniforme_manual <- function(x,max=1, min=0){

  rep(1/(max-min), length(x))
}
```

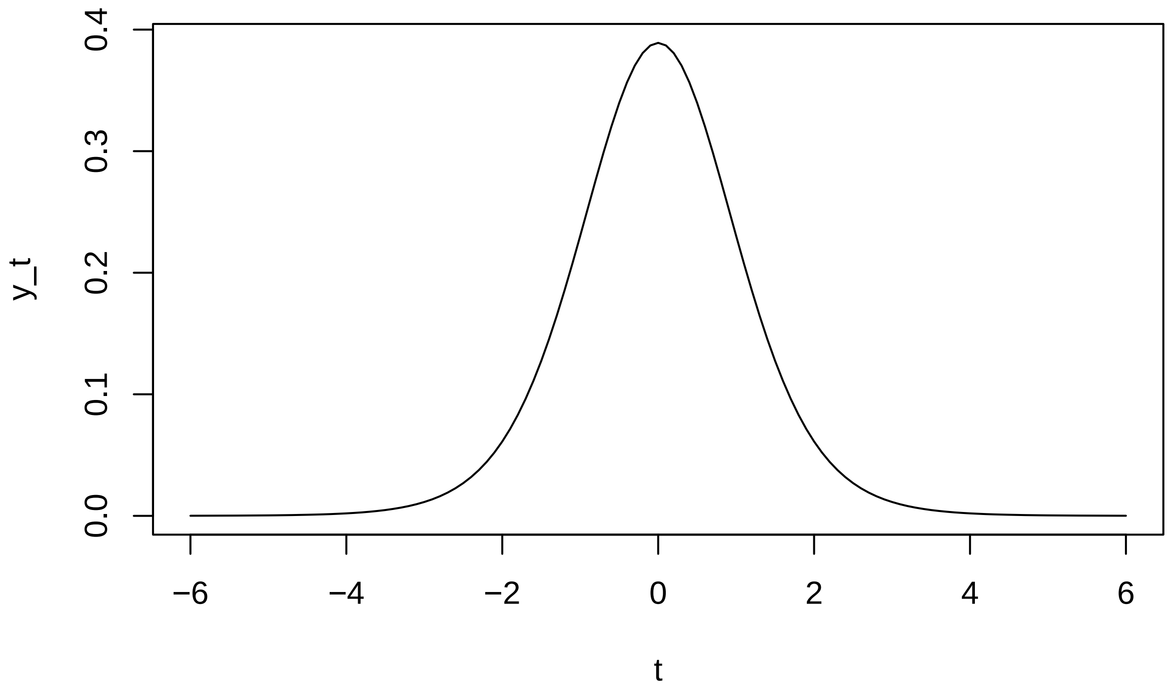
```r
x <- seq(-10,10,0.1)

y_generador <- uniforme_manual(x,1,0)

plot(y_generador~x,type="l")
```



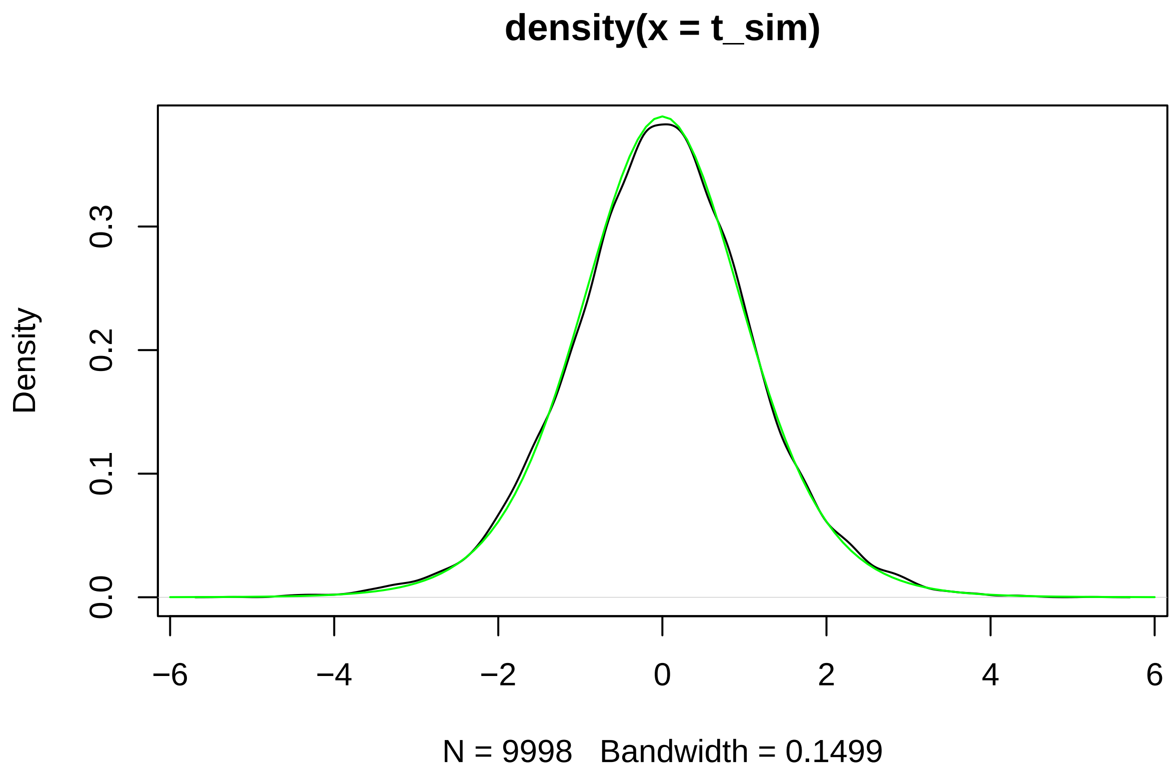

```r
#empirica

uniforme_sim <- runif(10000,0,1)

d <- density(uniforme_sim)

plot(d)
lines(y_generador~x, col="green")
```



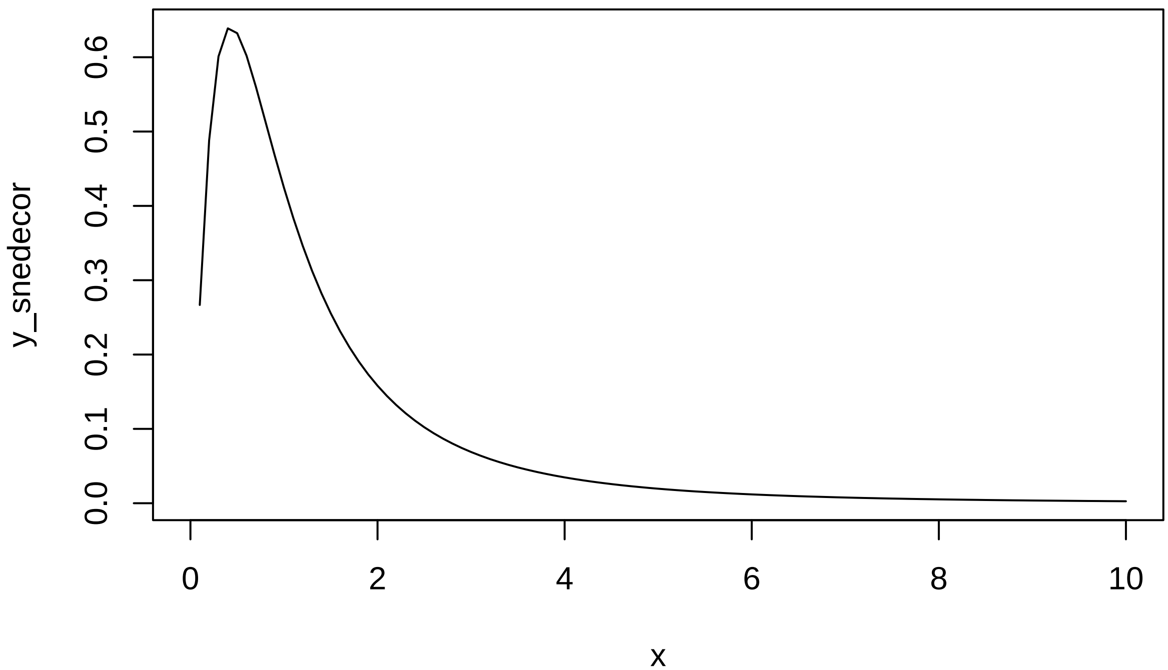

```r
### 2- DISTRIBUCION T-STUDENT

set.seed(2)

#teorica

t_manual <- function(t,gl){
  t=(gamma((gl+1)/2)/(sqrt(gl*pi)*gamma(gl/2)))*(1+t**2/gl)^-((gl+1)/2)
}


df <- 10
t <- seq(-6,6,0.1)

y_t <- t_manual(t,df)

plot(y_t~t, type="l")
```



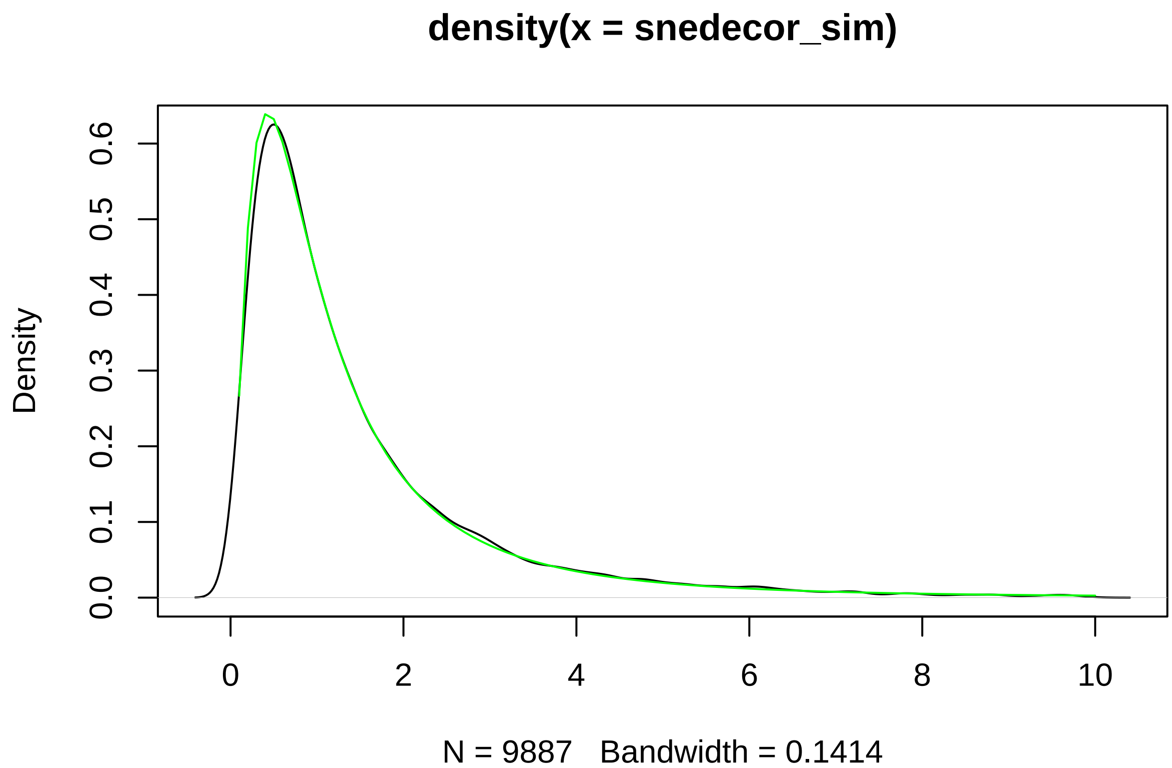

```r
#empirica

t_sim <- rt(10000,df)
t_sim <- t_sim[t_sim>-6 & t_sim<6]

d_t <- density(t_sim)

plot(d_t)
lines(y_t~t, col="green")
```



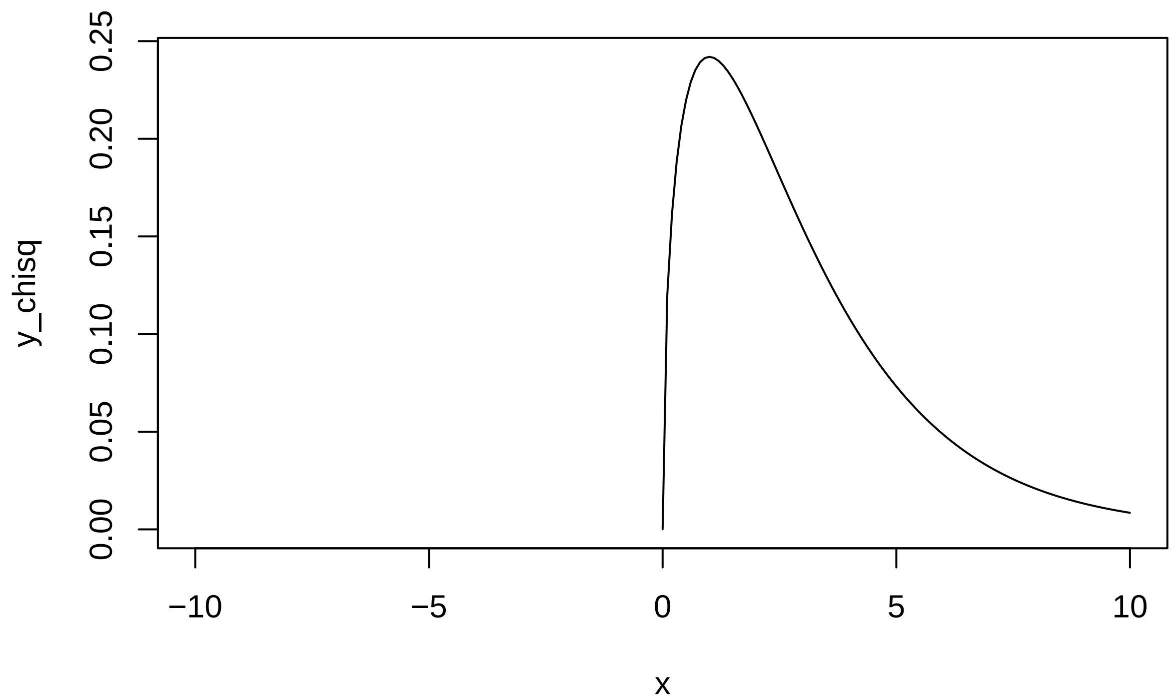

```r
### 3- DISTRIBUCION F-Snedecor

set.seed(3)

#teorica

snedecor_manual = function (x,m,n) {
  f = 1/(x*beta(m/2,n/2))*sqrt((m*x)^m*(n^n)/(m*x+n)^(m+n))
} #m y n son los grados de libertad

x <- seq(0,10,0.1)
m=5
n=5

y_snedecor <- snedecor_manual(x,m,n)

plot(y_snedecor~x, type="l")
```



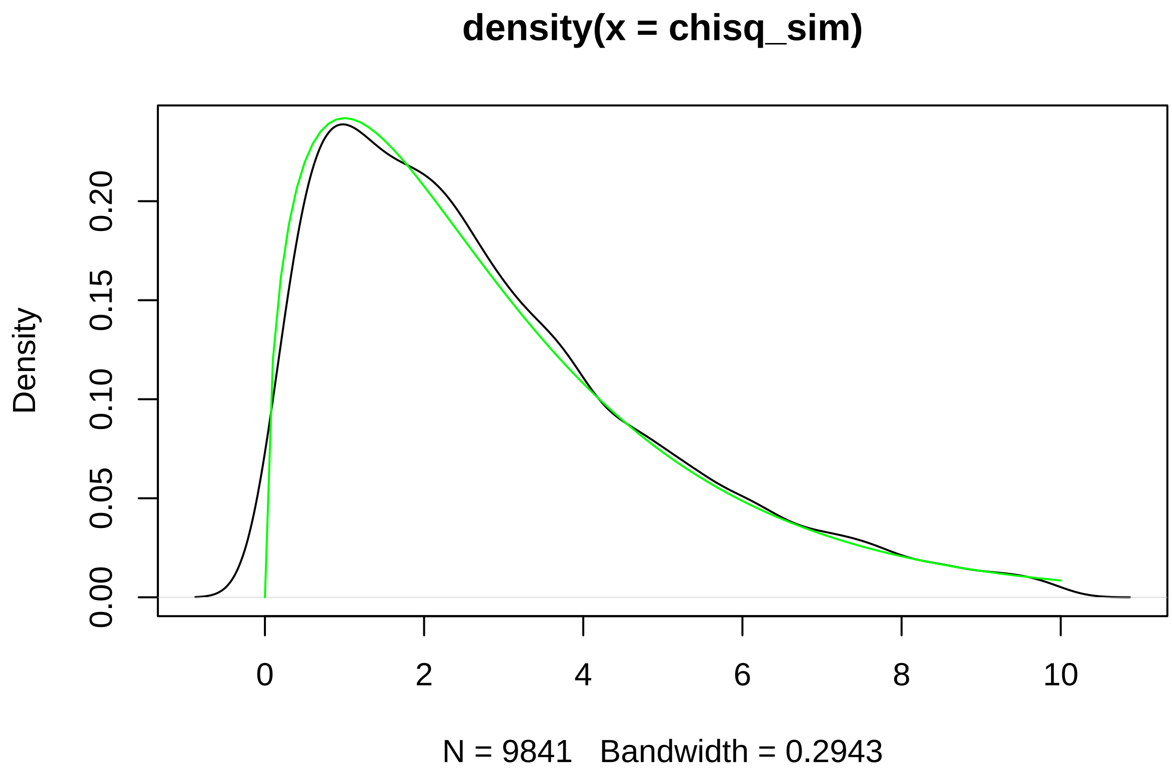

```
#empirica

snedecor_sim <- rf(10000,m,n)
snedecor_sim <- snedecor_sim[snedecor_sim>-10 & snedecor_sim<10]

d_snedecor <- density(snedecor_sim)
plot(d_snedecor)

lines(y_snedecor~x, col="green")
```




```r
### 4- DISTRIBUCION CHI-CUADRADO

set.seed(4)

#teorica

rchisq_manual = function(x, gl){
  n = (((1/2)^(gl/2)) / (gamma (gl/2))) * (x^((gl/2)-1)) * (exp(-x/2))
}

x <- seq(-10,10,0.1)
gl=3

y_chisq <- rchisq_manual(x,gl)

plot(y_chisq~x, type="l")
```




```r
#empirica

chisq_sim <- rchisq(10000,gl)

chisq_sim <- chisq_sim[chisq_sim>-10 & chisq_sim<10]

d_chisq <- density(chisq_sim)

plot(d_chisq)
lines(y_chisq~x,col="green")
```



2. EJERCICIO SOBRE LA LEY DÉBIL DE LOS GRANDES NÚMEROS

Muestra como el valor empírico y el teórico se aproximan al aumentar el n en el caso de tirar 5 dados y que salga el 1.

```
set.seed(5)

#teorica

k <- 5 #numero de dados

prob_ <- 5/6 #probabilidad de sacar cualquier numero que no sea 1

v_teorico <- 1-(prob_^k)  #probabilidad de que al menos un dado saque 1

v_teorico #(casi 0.6)
```

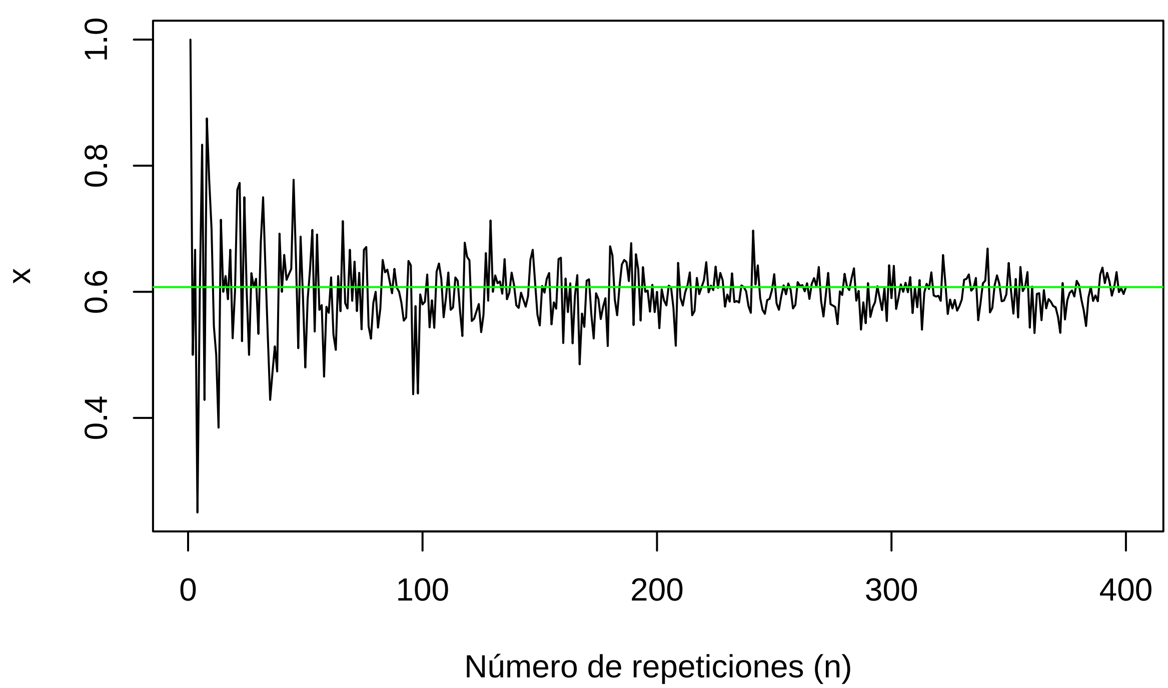
```
## [1] 0.5981224
```

```
#empirica (n=400)

prob <- 1/6
```

```r
x <- c()

for (n in 1:400) {

  # Simulamos lanzar 5 dados y verificamos si al menos uno muestra un 1

  resultados <- replicate(n, any(sample(1:6, 5, replace = TRUE) == 1))

  x[n] <- mean(resultados)
  # Calcula la proporción de veces que el resultado es TRUE, es decir, la proporción de simulaciones en
}

v_empirico <- mean(resultados)
v_empirico #0.6
```

```
## [1] 0.6075
```

```r
plot(x, type="l", xlab="Número de repeticiones (n)")

abline(h=v_empirico,col="green")
```



3. EJERCICIOS SOBRE EL TEOREMA CENTRAL DEL LÍMITE

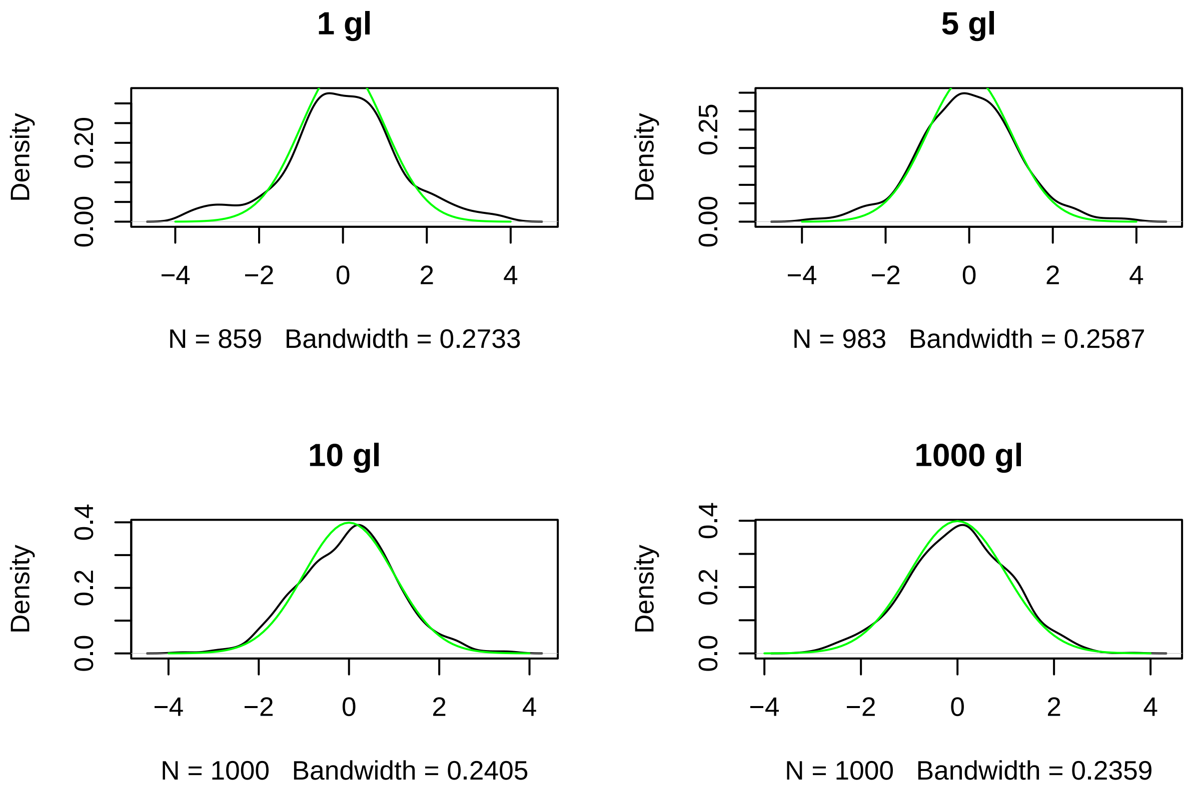
Una vez que hemos comprobado que R nos ofrece generadores aleatorios válidos podemos empezar a realizar las primeras simulaciones. La más sencilla, creo, es la del Teorema central de límite. Este teorema establece que cualquier distribución acaba por parecerse a la normal si se extrae suficiente muestra. Esto no es exactamente así y supone una gran simplificación, pero nos puede valer para no complicarnos.

El tamaño de la muestra en las distribuciones t y F afectan directamente a los grados de libertad, así que podíamos traducirlo como que al aumentar los grados de libertad de una distribución esta se aproxima a la normal.

Es decir, en teoría, al aumentar los grados de libertad (lo que indirectamente significa aumentar el tamaño muestral), las distribuciones se acercan a la normal. ¿Se comprobará empiricamente el teorema con las distribuciones t, F y χ^2?

a) Muestre el teorema central del límite con la distribución T

```
#Distribucion T

normal <- function(x, med, sd){
  n = (1/sqrt(2*pi*sd**2)) * exp(-(x-med)**2/(2*sd**2))
}
recorta <- function(y, li, ls){
  return(y[y>li & y<ls])
}
x <- seq(-4, 4, .1) # los valores estaran en este rango
y.normal <- normal(x, 0, 1) # distribucion teorica normal

y.tgl1 <- rt(1000, 1) # distribuciones generadas por rt() con gl = 1
y.tgl5 <- rt(1000, 5) # distribuciones generadas por rt() con gl = 5
y.tgl10 <- rt(1000, 10) # distribuciones generadas por rt() con gl = 10
y.tgl100 <- rt(1000, 100) # distribuciones generadas por rt() con gl = 100

# recortamos para simplificar
y.tgl1 <- recorta(y.tgl1, -4, 4)
y.tgl5 <- recorta(y.tgl5, -4, 4)
y.tgl10 <- recorta(y.tgl10, -4, 4)
y.tgl100 <- recorta(y.tgl100, -4, 4)

par(mfrow=c(2,2))

plot(density(y.tgl1), main="1 gl")
lines(y.normal~x, col="green")

plot(density(y.tgl5), main="5 gl")
lines(y.normal~x, col="green")
plot(density(y.tgl10), main="10 gl")
lines(y.normal~x, col="green")

plot(density(y.tgl100), main="1000 gl")
lines(y.normal~x, col="green")
```

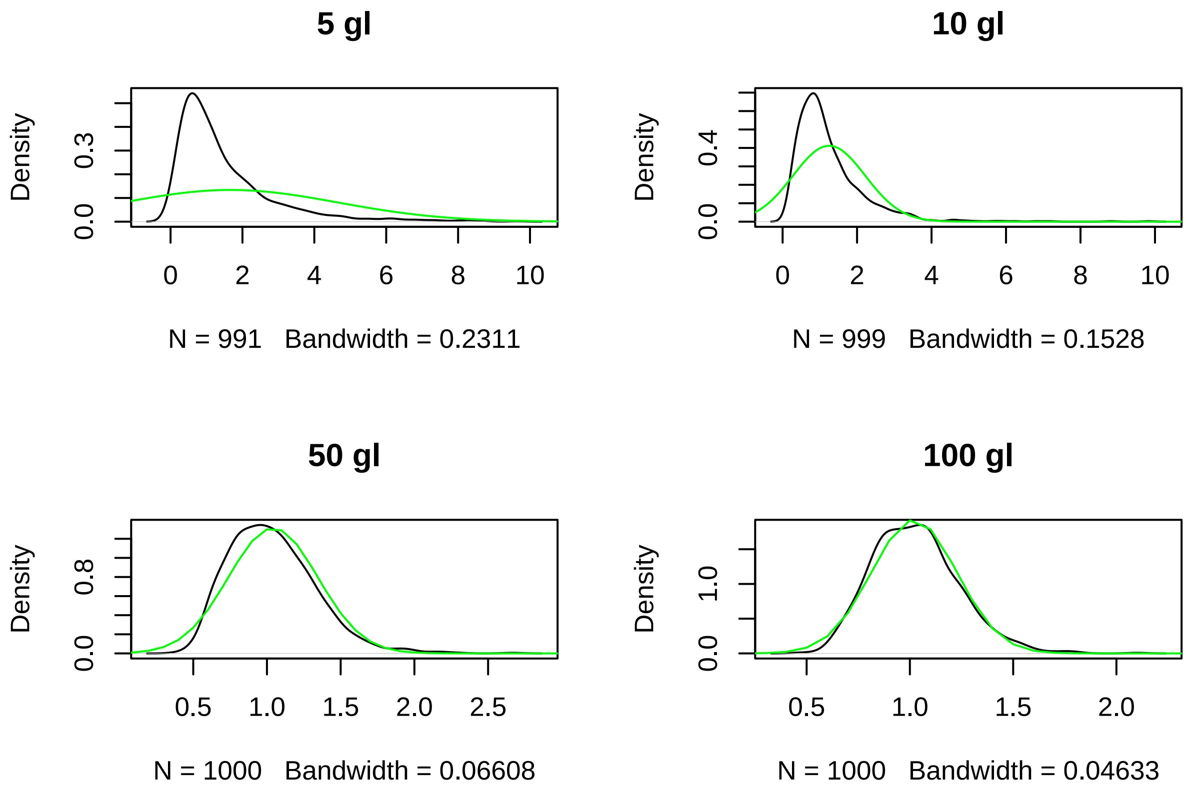


Las figuras anteriores aparecen perfectamente centradas la distribución normal y la t porque ambas tienen la media y la moda en cero, pero esto no ocurre con la distribución F cuya media es: $\frac{gl_2}{(gl_2-2)}$ para valores de $gl_2 > 2$ ni con valores de χ^2 cuya media es gl (sus grados de libertad). Por ello hay que obtener una normal equivalente con media y dispersion de cada distribucion F y χ^2 en funcion de sus grados de libertad.

b) Muestre el teorema central del límite con la distribución F. Como F tiene gl_1 y gl_2, para no hacer infinidad de gráficos, haga la demostración para $gl_1 = gl_2$, es decir, siempre los mismos grados de libertad en el numerador y el denominador.

```
#Distribucion F


# DISTRIBUCION F

set.seed(5)

media_F = function(gl){
  return(gl / (gl-2))
}

sd_F = function(gl){
  sqrt((2*(gl**2)*(gl+gl-2))/(gl*((gl-2)**2)*(gl-4)))
}
```

```r
x = seq(-100, 200, .1) # los valores estaran en este rango

y_fgl5 <- rf(1000,5,5)
y_fgl10 <- rf(1000,10,10)
y_fgl50 <- rf(1000,50,50)
y_fgl100 <- rf(1000,100,100)

#distribuciones normales según los grados de libertad

y.ecuacion5 = dnorm(x, media_F(5), sd_F(5))
y.ecuacion10 = dnorm(x, media_F(10), sd_F(10))
y.ecuacion50 = dnorm(x, media_F(50), sd_F(50))
y.ecuacion100 = dnorm(x, media_F(100), sd_F(100))


y_fgl5 <- recorta(y_fgl5,-10,10)
y_fgl10 <- recorta(y_fgl10,-10,10)
y_fgl50 <- recorta(y_fgl50,-10,10)
y_fgl100 <- recorta(y_fgl100,-10,10)


par(mfrow=c(2,2))

plot(density(y_fgl5), main="5 gl")
lines(y.ecuacion5~x, col="green")

plot(density(y_fgl10), main="10 gl")
lines(y.ecuacion10~x, col="green")

plot(density(y_fgl50), main="50 gl")
lines(y.ecuacion50~x, col="green")

plot(density(y_fgl100), main="100 gl")
lines(y.ecuacion100~x, col="green")
```



c) Muestre el teorema central del límite con la distribución χ^2

En este caso la media de χ^2 son los grados de libertad y la desviacion tipica es $\sqrt{(}2 * gl)$

```
# Distribucion chi-cuadrado


set.seed(5)

x = seq(-100, 200, .1)

y.chiqgl1 = rchisq(1000, 1)
y.chiqgl5 = rchisq(1000, 5)
y.chiqgl10 = rchisq(1000, 10)
y.chiqgl100 = rchisq(1000, 100)

#distribuciones normales según los grados de libertad

y.ecuacion1 = dnorm(x, 1, sqrt (2*1))
y.ecuacion5 = dnorm(x, 5, sqrt (2*5))
y.ecuacion10 = dnorm(x, 10, sqrt (2*10))
y.ecuacion100 = dnorm(x, 100, sqrt (2*100))

y.chiqgl1 = recorta(y.chiqgl1, -2, 10)
y.chiqgl5 = recorta(y.chiqgl5, 0, 10)
```

```r
y.chiqgl10 = recorta(y.chiqgl10, 0, 15)
y.chiqgl100 = recorta(y.chiqgl100, 0, 150)

par(mfrow=c(2,2))

plot(density(y.chiqgl1), main="1 gl")
lines(y.ecuacion1~x, col="green")

plot(density(y.chiqgl5), main="5 gl")
lines(y.ecuacion5~x, col="green")

plot(density(y.chiqgl10), main="10 gl")
lines(y.ecuacion10~x, col="green")

plot(density(y.chiqgl100), main="100 gl")
lines(y.ecuacion100~x, col="green")
```