03.Distribuciones_Muestrales_Estadisticos

Alicia Gil Matute

2024-09-20

Contents

Estudio y manipulación de la normalidad de una variable

Para estudiar la normalidad de una variable se suelen utilizar dos estrategias, la primera consiste en describir la variable y la segunda en realizar algún contraste de hipótesis. Los descriptivos básicos usados son los momentos de Pearson (media, desviación asimetría y apuntamiento), los histogramas, las figuras de caja y bigote y, lo más útil, los qqplot. Para contrastar si una variable es normal se pueden usar los contrastes de Kolmogorov-Smirnov o de Shapiro-Wilk.

Cuando una variable no es normal se puede intentar normalizar usando algún tipo de transformación como la de Box-Cox que funciona razonablemente bien. Una segunda situación ligada a esta es justo la inversa: mi distribución es normal pero quiero que tenga la forma que yo desee, con los 4 momentos (media, desviación, asimetría y apuntamiento) definidos por mí. Esto es lo que me permite hacer la distribución de Johnson, implementada en el paquete SuppDists.

Un aviso muy importante: No todas las combinaciones de parámetros son posibles, a veces no se pueden dibujar y a veces no producen exactamente el resultado deseado. Esto es debido a que matemáticamente no se pueden definir inconsistencias entre los momentos de una distribución. Cuando esto ocurra simplemente probamos configuraciones de parámetros más razonables.

Tarea 1

Utilizando la distribución de Johnson calcula por fuerza bruta qué combinaciones de parámetros generan una distribución válida. Puedes fijar los valores de la media y la desviación típica a 0 y 1 y probar con un doble bucle. Valores razonables de asimetría y apuntamiento para probar pueden ser [-2, 2] y [0,6] respectivamente. Recuerda que las asimetría y el apuntamiento (tercer y cuarto momento de pearson) de la normal son 0 y 3 respectivamente.

Cuando una función falla en R se detiene el procesamiento pero eso es necesario utilizar funciones como tryCatch. Busca ejemplos de su uso.

Tarea 2

Utilizando la distribución de Johnson dibuja las siguientes distribuciones:

- Una normal
- Una platicúrtica
- Una mesocúrtica
- Una asimétrica positiva
- Una asimétrica negativa

```
# Una normal

set.seed(3)

datos_n <- tryCatch({JohnsonFit(c(0,1,0,3), moment="use")}, error=function(e){return(print(0))})

normal_johnson <- rJohnson(1000,datos_n)
sJohnson(normal_johnson)
```

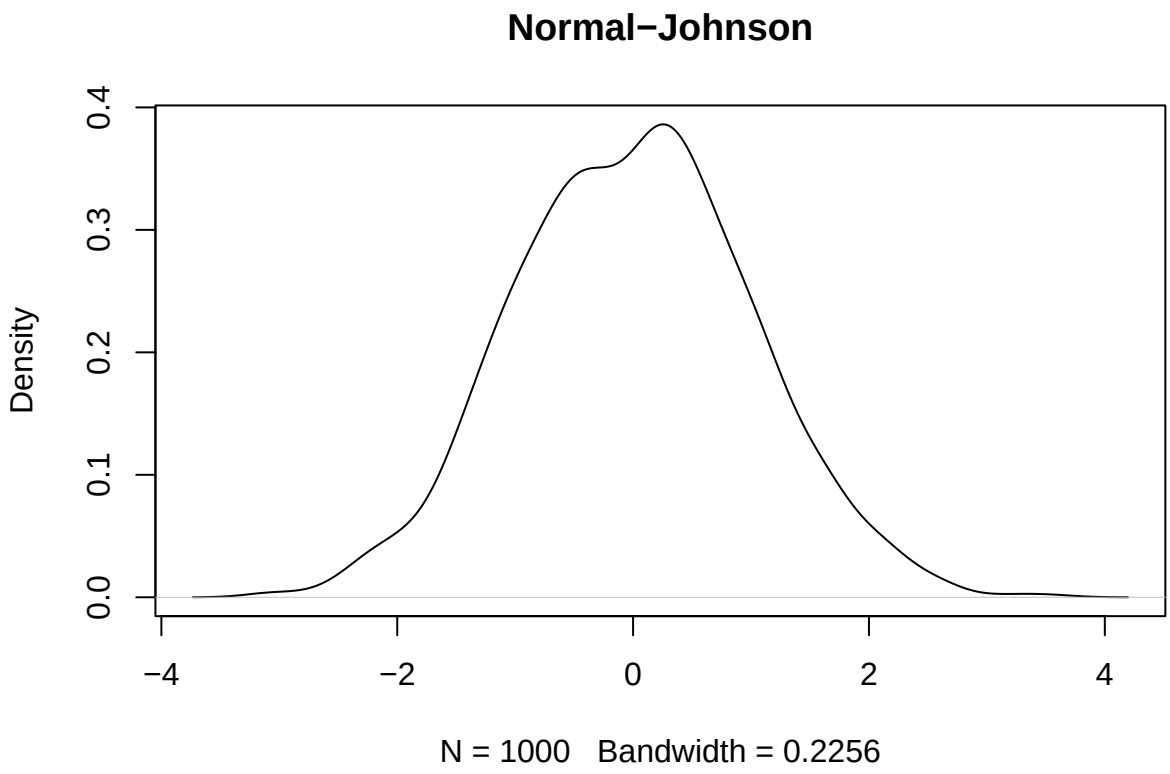
```
## $title
## [1] "Johnson Distribution"
##
## $gamma
## [1] -0.9619334
##
## $delta
## [1] -0.2925257
##
## $xi
## [1] 0.2587882
##
## $lambda
## [1] -1.152132
##
## $type
## [1] 4
##
## $Mean
## [1] 0.03331633
```

```
##
## $Median
## [1] 0.2173434
##
## $Mode
## [1] 0.2587871
##
## $Variance
## [1] 0.1135763
##
## $SD
## [1] 0.3370108
##
## $ThirdCentralMoment
## [1] -0.05851769
##
## $FourthCentralMoment
## [1] 0.05149116
##
## $PearsonsSkewness...mean.minus.mode.div.SD
## [1] -0.6690313
##
## $Skewness...sqrtB1
## [1] -1.528818
##
## $Kurtosis...B2.minus.3
## [1] 0.9916955
```

```
shapiro.test(normal_johnson)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  normal_johnson
## W = 0.9991, p-value = 0.9206
```

```
plot(density(normal_johnson), main="Normal-Johnson")
```

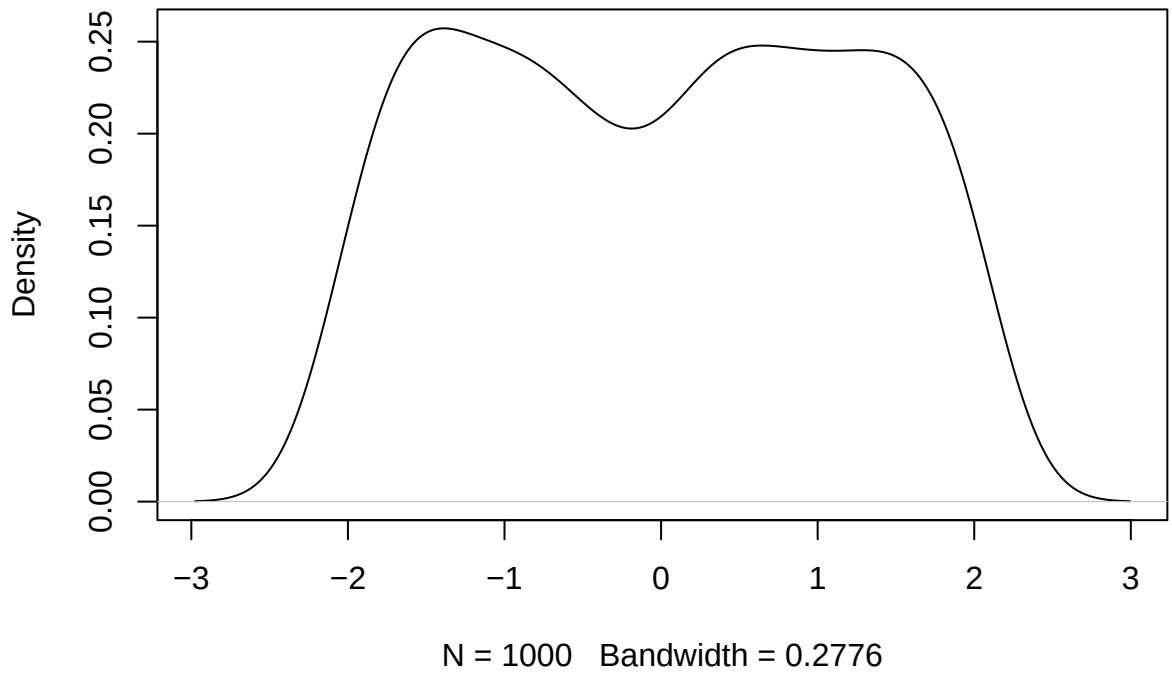



```
#Una platicurtica = kurtosis<0

set.seed(3)

datos_p <- tryCatch({JohnsonFit(c(0,1.5,0,4), moment="use")}, error=function(e){return(print(0))})

p_johnson <- rJohnson(1000,datos_p)
sJohnson(p_johnson)
```

```
## $title
## [1] "Johnson Distribution"
##
## $gamma
## [1] -1.409192
##
## $delta
## [1] -0.5010551
##
## $xi
## [1] 0.4449954
##
## $lambda
## [1] -1.583394
##
## $type
```

```
## [1] 4
##
## $Mean
## [1] 0.2168286
##
## $Median
## [1] 0.355288
##
## $Mode
## [1] 0.4327505
##
## $Variance
## [1] 0.09676463
##
## $SD
## [1] 0.3110701
##
## $ThirdCentralMoment
## [1] -0.05869321
##
## $FourthCentralMoment
## [1] 0.05964667
##
## $PearsonsSkewness...mean.minus.mode.div.SD
## [1] -0.6941261
##
## $Skewness...sqrtB1
## [1] -1.949902
##
## $Kurtosis...B2.minus.3
## [1] 3.370198
```

```r
plot(density(p_johnson), main="Platicúrtica-Johnson")
```

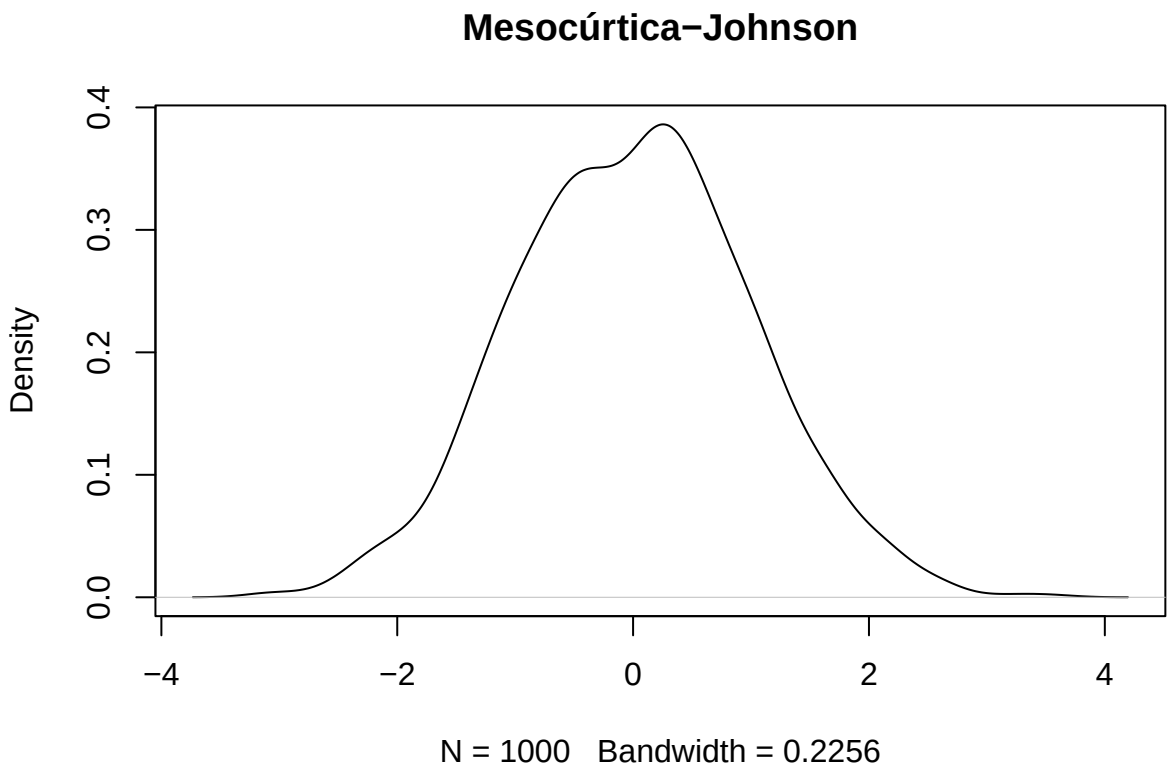



```
#Una mesocúrtica. La normal es mesocurtica

set.seed(3)

datos_m <- tryCatch({JohnsonFit(c(0,1,0,3), moment="use")}, error=function(e){return(print(0))})

m_johnson <- rJohnson(1000,datos_m)
sJohnson(m_johnson)
```

```
## $title
## [1] "Johnson Distribution"
##
## $gamma
## [1] -0.9619334
##
## $delta
## [1] -0.2925257
##
## $xi
## [1] 0.2587882
##
## $lambda
## [1] -1.152132
##
## $type
```

```
## [1] 4
##
## $Mean
## [1] 0.03331633
##
## $Median
## [1] 0.2173434
##
## $Mode
## [1] 0.2587871
##
## $Variance
## [1] 0.1135763
##
## $SD
## [1] 0.3370108
##
## $ThirdCentralMoment
## [1] -0.05851769
##
## $FourthCentralMoment
## [1] 0.05149116
##
## $PearsonsSkewness...mean.minus.mode.div.SD
## [1] -0.6690313
##
## $Skewness...sqrtB1
## [1] -1.528818
##
## $Kurtosis...B2.minus.3
## [1] 0.9916955
```

```
plot(density(m_johnson), main="Mesocúrtica-Johnson")
```

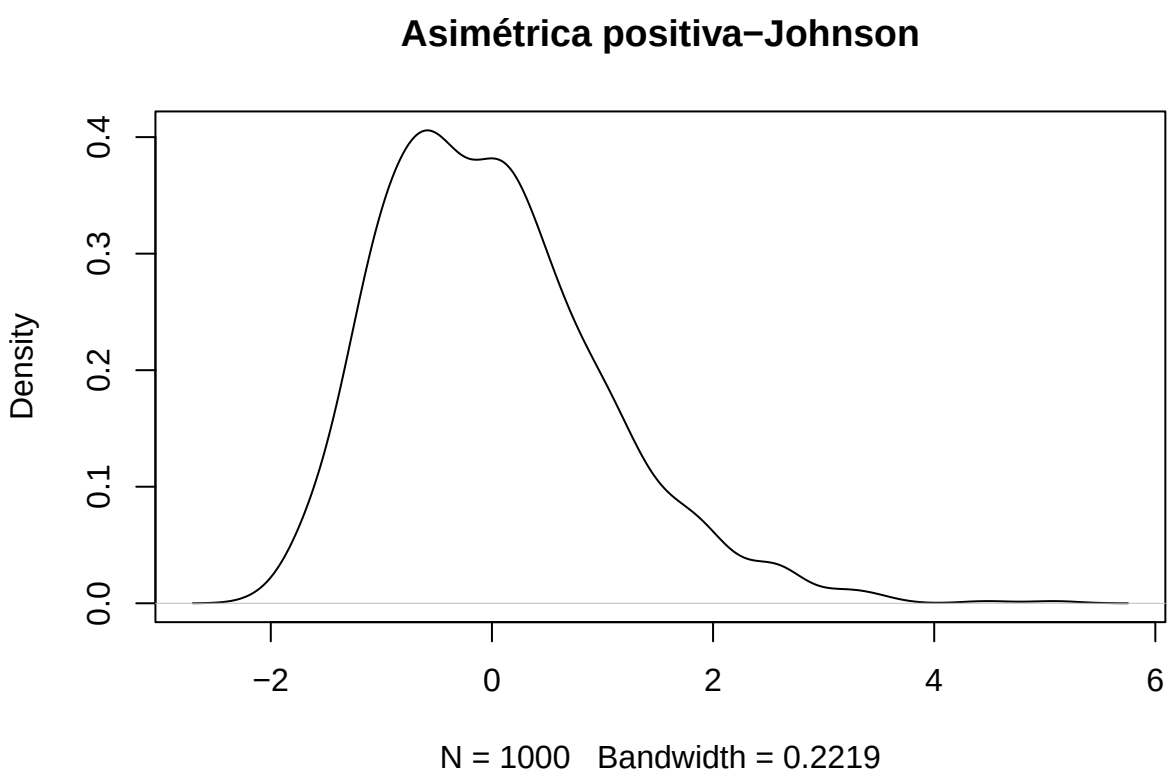



```r
#Una asimétrica positiva

set.seed(3)

datos_asiN <- tryCatch({JohnsonFit(c(0,1,.8,4), moment="use")}, error=function(e){return(print(0))})

asiN_johnson <- rJohnson(1000,datos_asiN)
sJohnson(asiN_johnson)
```

```
## $title
## [1] "Johnson Distribution"
##
## $gamma
## [1] -0.9405777
##
## $delta
## [1] -0.4030494
##
## $xi
## [1] 0.1261041
##
## $lambda
## [1] -1.074219
##
## $type
```

```
## [1] 4
##
## $Mean
## [1] -0.110489
##
## $Median
## [1] 0.0311713
##
## $Mode
## [1] 0.1260554
##
## $Variance
## [1] 0.08564383
##
## $SD
## [1] 0.2926497
##
## $ThirdCentralMoment
## [1] -0.03338645
##
## $FourthCentralMoment
## [1] 0.02636436
##
## $PearsonsSkewness...mean.minus.mode.div.SD
## [1] -0.8082853
##
## $Skewness...sqrtB1
## [1] -1.332067
##
## $Kurtosis...B2.minus.3
## [1] 0.5943895
```

```
plot(density(asiN_johnson), main="Asimétrica positiva-Johnson")
```

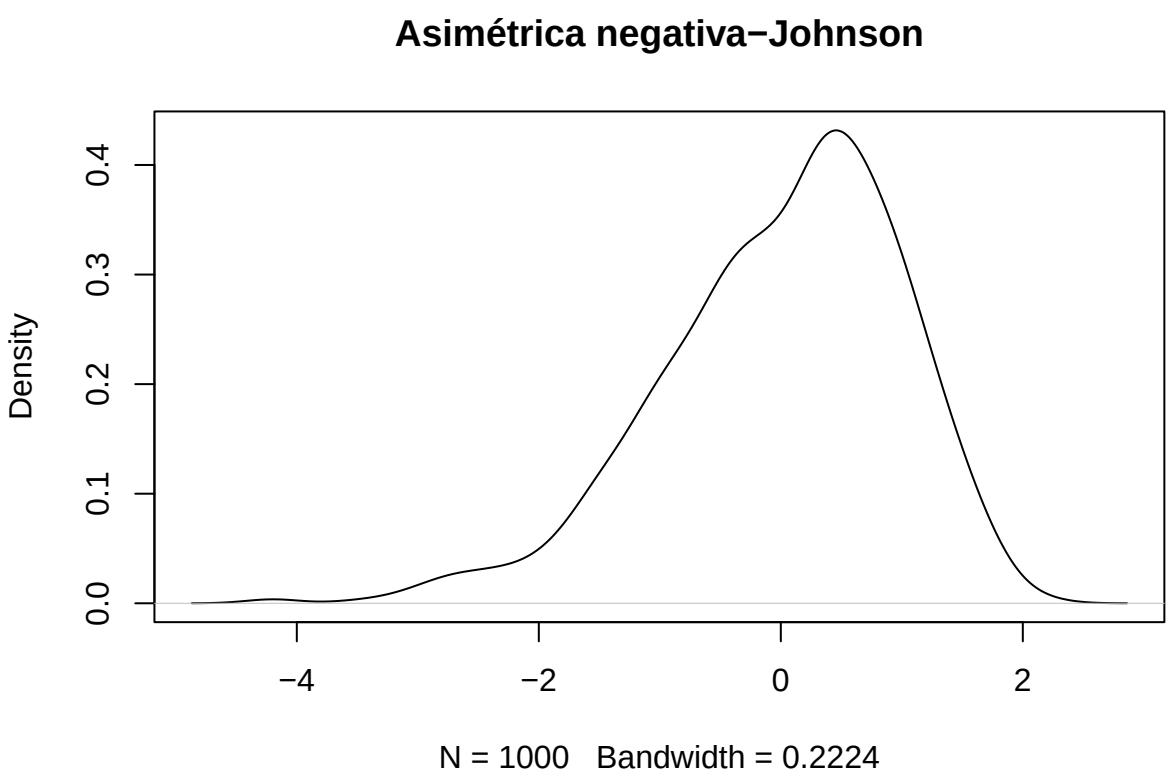



```
#Una asimétrica negativa

set.seed(3)

datos_asiP <- tryCatch({JohnsonFit(c(0,1,-.8,4), moment="use")}, error=function(e){return(print(0))})

asiP_johnson <- rJohnson(1000,datos_asiP)
sJohnson(asiP_johnson)
```

```
## $title
## [1] "Johnson Distribution"
##
## $gamma
## [1] -0.9267087
##
## $delta
## [1] -0.1612134
##
## $xi
## [1] 0.3730097
##
## $lambda
## [1] -1.168862
##
## $type
```

```
## [1] 4
##
## $Mean
## [1] 124.7951
##
## $Median
## [1] 0.3692952
##
## $Mode
## [1] 0.3730097
##
## $Variance
## [1] 5188442
##
## $SD
## [1] 2277.815
##
## $ThirdCentralMoment
## [1] -645560096
##
## $FourthCentralMoment
## [1] 80322350565
##
## $PearsonsSkewness...mean.minus.mode.div.SD
## [1] 0.05462342
##
## $Skewness...sqrtB1
## [1] -0.05462371
##
## $Kurtosis...B2.minus.3
## [1] -2.997016
```

```
plot(density(asiP_johnson), main="Asimétrica negativa-Johnson")
```



Distribución muestral de la media

Una distribución muestral de un estadístico es la distribución que adopta el estadístico al extraerse infinitas muestras de tamaño n de una población. Por ejemplo, para la media:

- Tenemos una población de 1000 elmentos (N=1000)
- Extraigo de ella 25000 muestras (k=25000) de 25 elementos (n=25), obviamente con reposición.
- Con esas k muestras calculo la media.
- Con esas k medias dibujo la distribución (histograma).

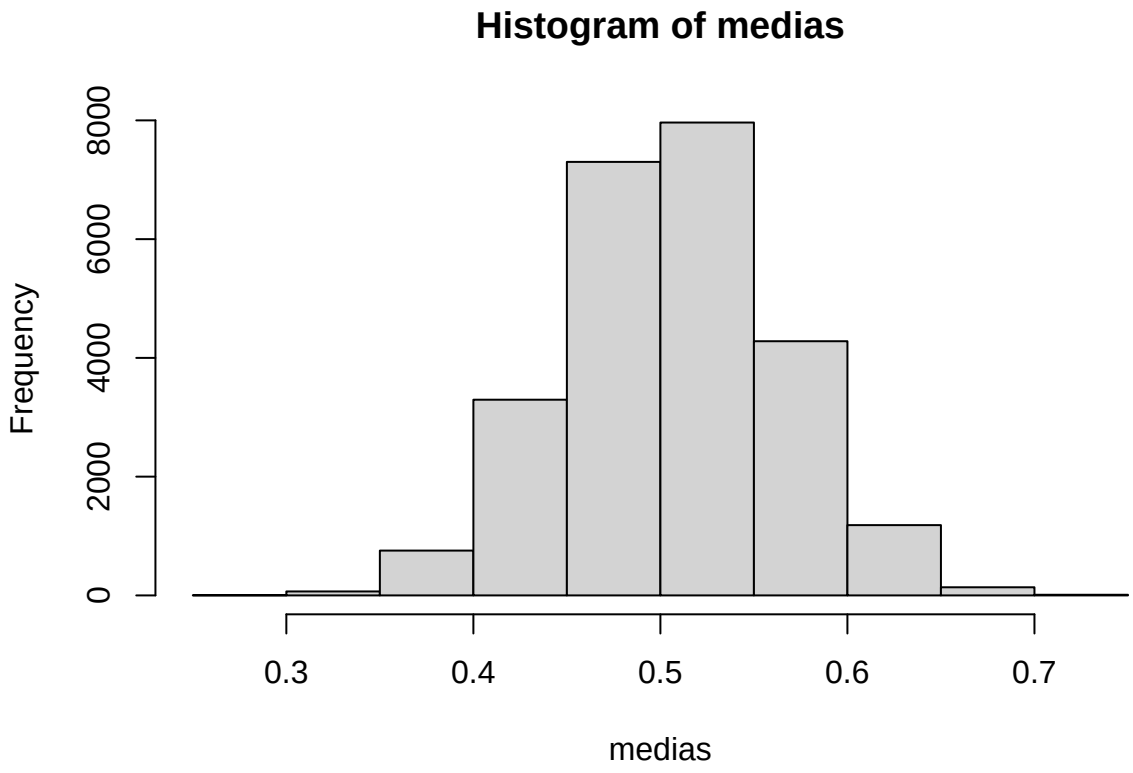
Si hacemos esto, la teoría nos dice que la distribución resultante, llamada **distribución muestral de la media**, cumple las siguientes condiciones:

- Tendrá forma normal, es decir $N(\mu, \frac{\sigma}{\sqrt(n)})$
- Su media es la media poblacional $\mu_{\bar{X}} = \mu$
- Su desviación típica es igual a $\sigma_{\bar{X}} = \frac{\sigma}{\sqrt(n)}$

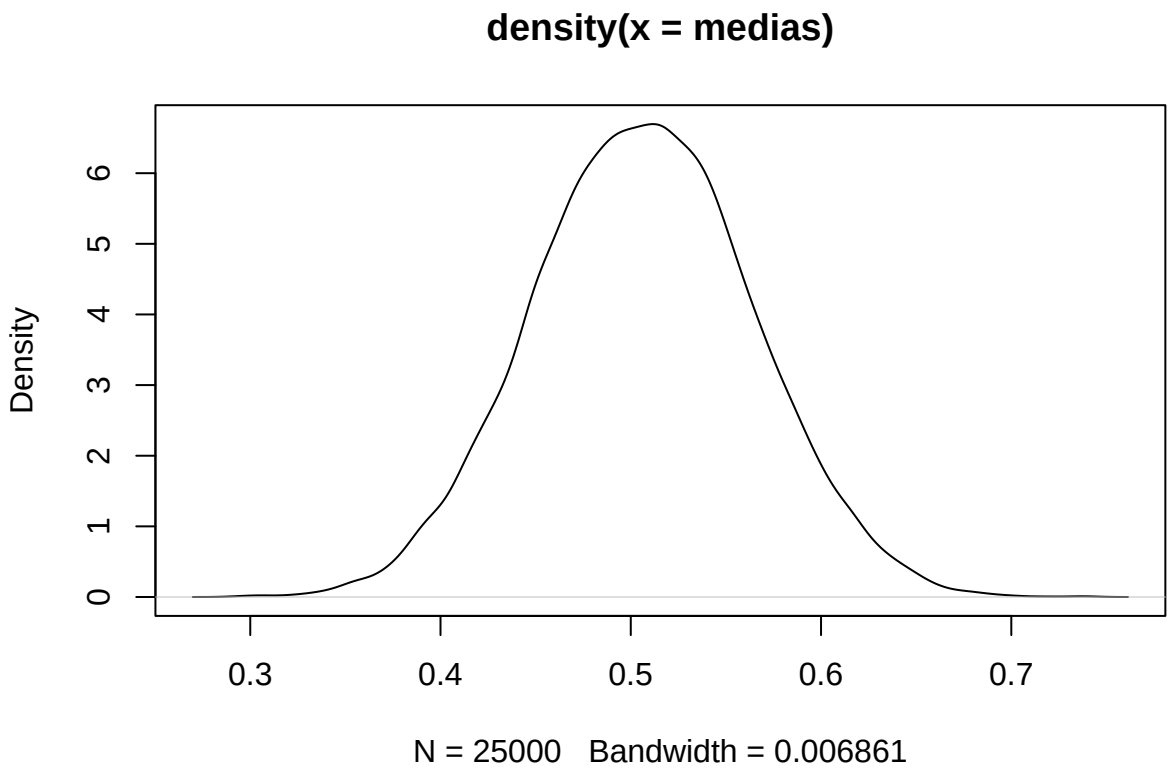
Tarea 3

Muestra que esto se cumple, aunque la población de origen no sea normal, prueba con la distribución uniforme y con una muy sesgada positiva como la χ^2.

```r
set.seed(3)

#distribucion uniforme

N=1000
k=25000
n=25

poblacion_unif <- runif(N,0,1)

mu_pob_u <- mean(poblacion_unif)

sd_pob_u <- sd(poblacion_unif)

medias <- c()

for (i in 1:k){

  muestra <- sample(poblacion_unif,n, replace=T)

  medias[i] <- mean(muestra)

}

hist(medias)
```



```r
plot(density(medias))
```




```r
#Comprobacion teórica vs empirica

mu_teo_u <- mu_pob_u

sd_teo_u <- sd_pob_u/sqrt(n)


mu_emp_u <- mean(medias)
sd_emp_u <- sd(medias)

sprintf("Media teórica: %.2f -- Media empírica: %.2f", mu_teo_u, mu_emp_u)
```

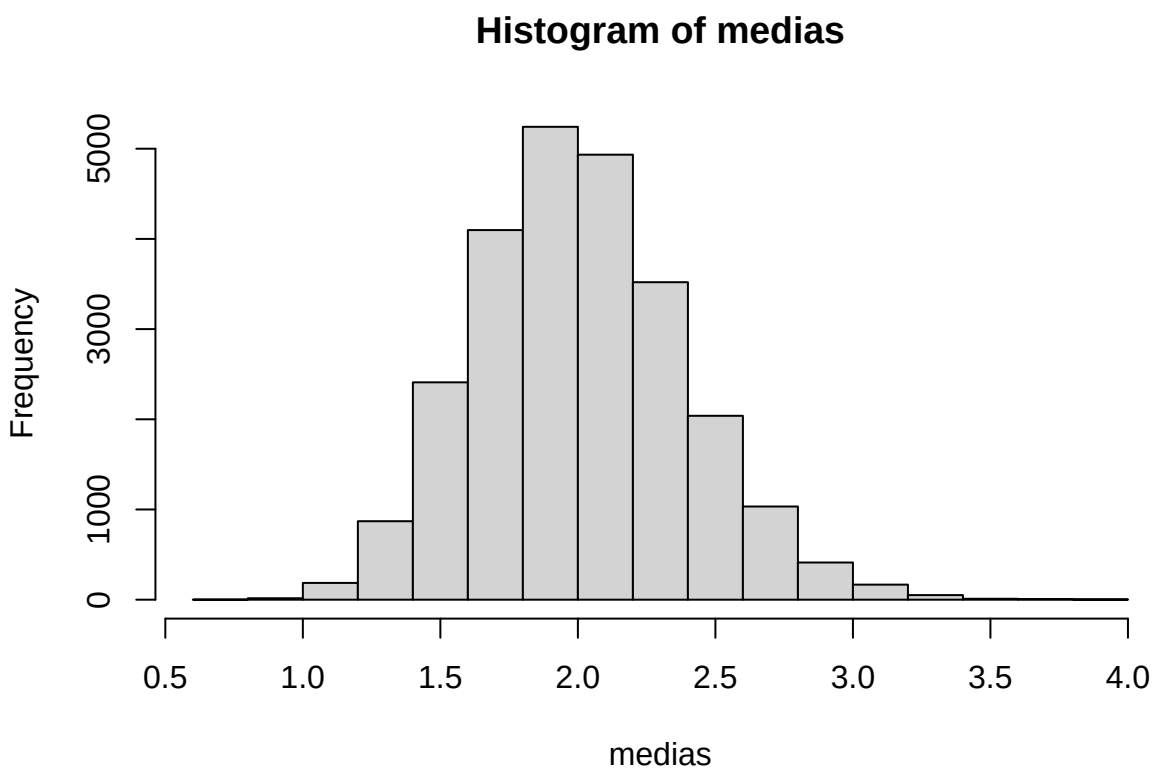
```
## [1] "Media teórica: 0.51 -- Media empírica: 0.51"
```

```r
sprintf("Sd teórica: %.2f -- Sd empírica: %.2f", sd_teo_u, sd_emp_u)
```

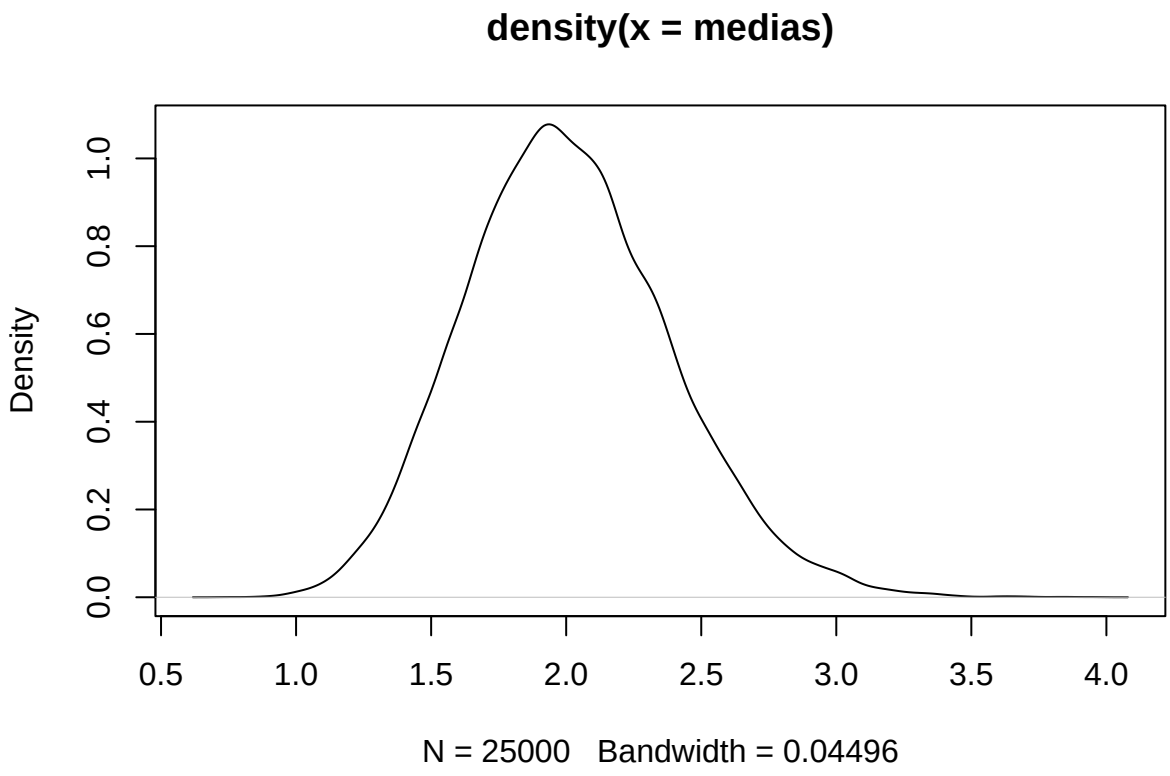
```
## [1] "Sd teórica: 0.06 -- Sd empírica: 0.06"
```

```r
# Distribucion chi-cuadrado (2 grados de libertad)

set.seed(3)
```

```r
N=1000
k=25000
n=25

poblacion_chi <- rchisq(1000,2)

mu_pob_chi <- mean(poblacion_chi)
sd_pob_chi <- sd(poblacion_chi)

medias <- c()

for (i in 1:k){

  muestras <- sample(poblacion_chi,n,replace=T)
  medias[i] <- mean(muestras)
}

hist(medias)
```




```r
plot(density(medias))
```




```
#Comprobación teórica vs empírica

mu_teo_chi <- mu_pob_chi
sd_teo_chi <- sd_pob_chi/sqrt(n)

mu_emp_chi <- mean(medias)
sd_emp_chi <- sd(medias)

sprintf("Media teórica: %.2f -- Media empírica: %.2f", mu_teo_chi, mu_emp_chi)
```

```
## [1] "Media teórica: 2.00 -- Media empírica: 2.01"
```

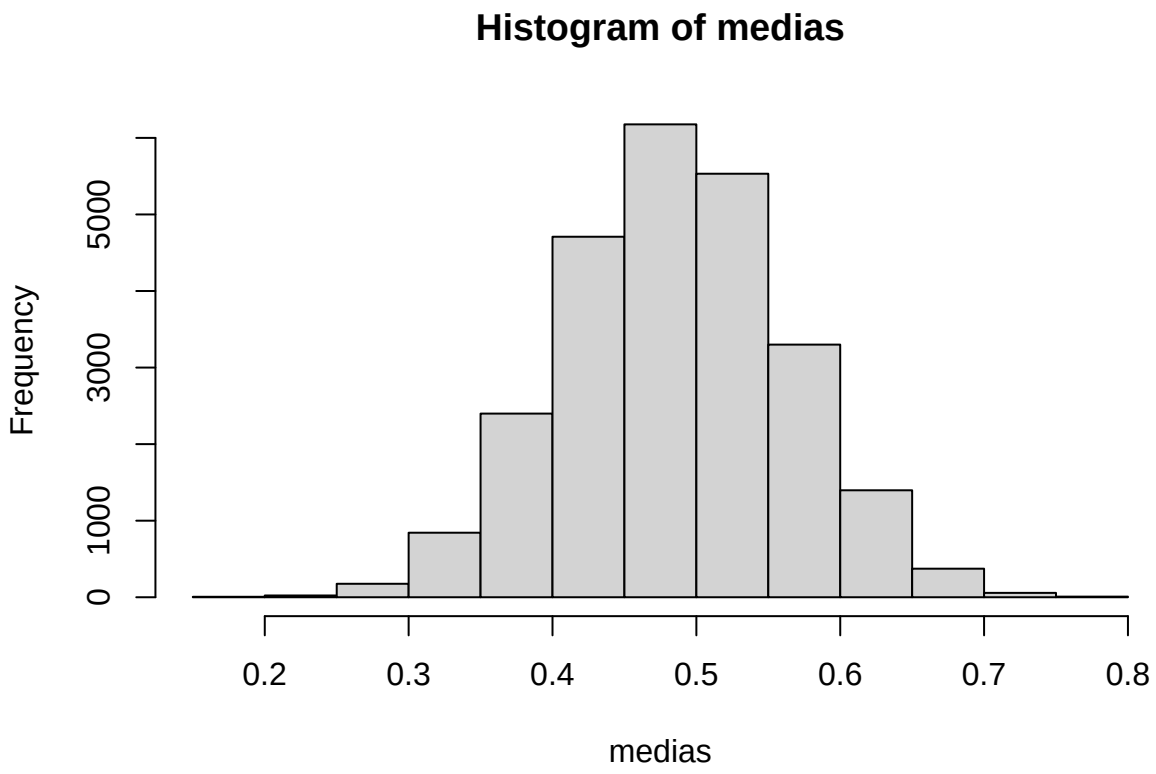
```
sprintf("Sd teórica: %.2f -- Sd empírica: %.2f", sd_teo_chi, sd_emp_chi)
```

```
## [1] "Sd teórica: 0.38 -- Sd empírica: 0.38"
```

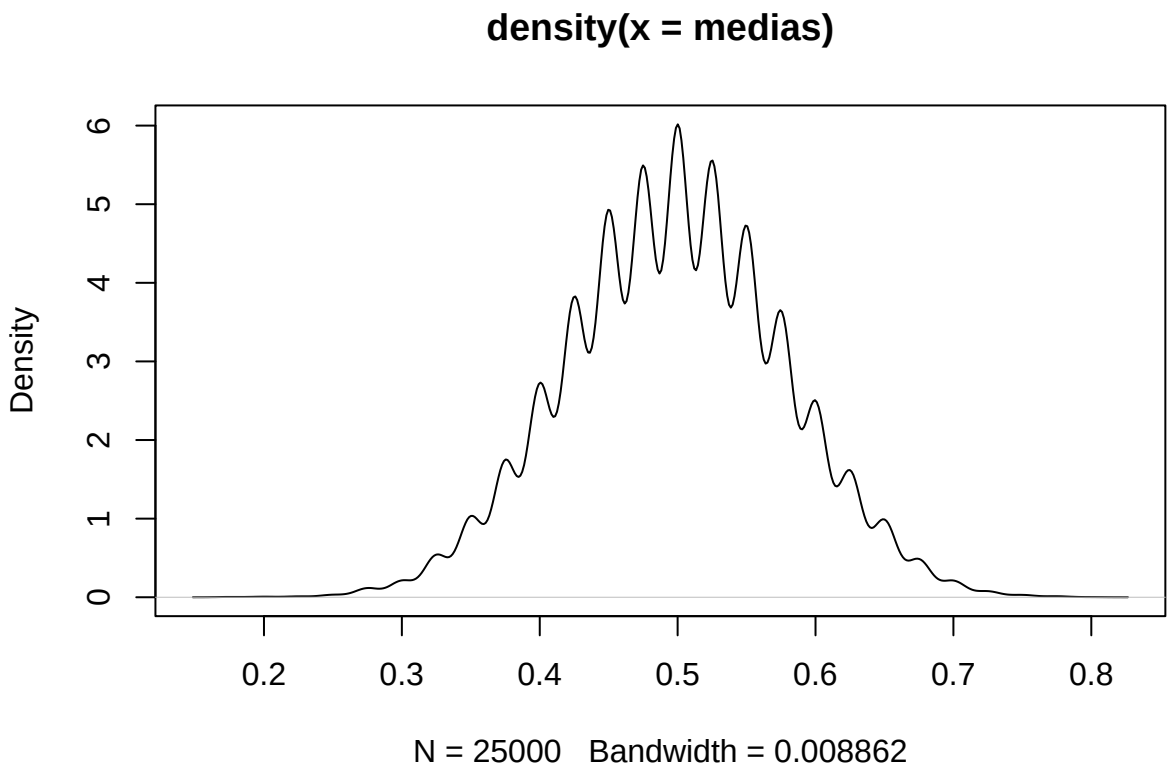
Tarea 4

Muestra que la distribución muestral de la proporción es también $N(\mu, \frac{\sigma}{\sqrt(n)})$ con muestras de aproximadamente $n \geq 30$, pero no lo es para muestras más pequeñas (por ejemplo de n=5). Recuerda que la población que tienes que crear ahora es binomial (ceros y unos).

```r
set.seed(3)

N=1000
k=25000

#con n>30

n1=40

x <- seq(0,100,0.01)

poblacion_n1 <- rbinom(x,1,0.5)

mu_pob_binom <- mean(poblacion_n1)
sd_pob_binom <- sd(poblacion_n1)

medias <- c()

for (i in 1:k){

  muestras <- sample(poblacion_n1,n1,replace = T)
  medias[i] <- mean(muestras)

}

hist(medias)
```




```
plot(density(medias))
```




```
#Comprobación teórica vs empírica

mu_teo_binom <- mu_pob_binom
sd_teo_binom <- sd_pob_binom/sqrt(n1)

mu_emp_binom <- mean(medias)
sd_emp_binom <- sd(medias)

sprintf("Media teórica: %.2f -- Media empírica: %.2f", mu_teo_binom, mu_emp_binom)
```

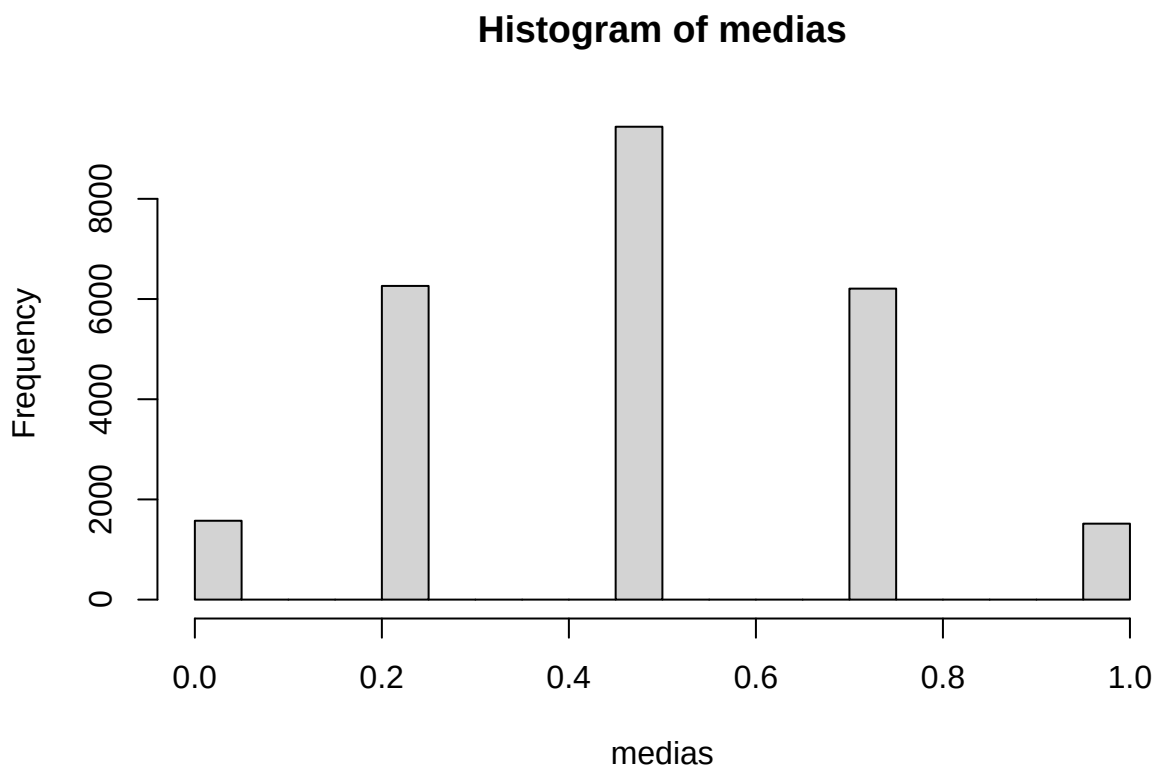
```
## [1] "Media teórica: 0.50 -- Media empírica: 0.50"
```

```
sprintf("Sd teórica: %.2f -- Sd empírica: %.2f", sd_teo_binom, sd_emp_binom)
```

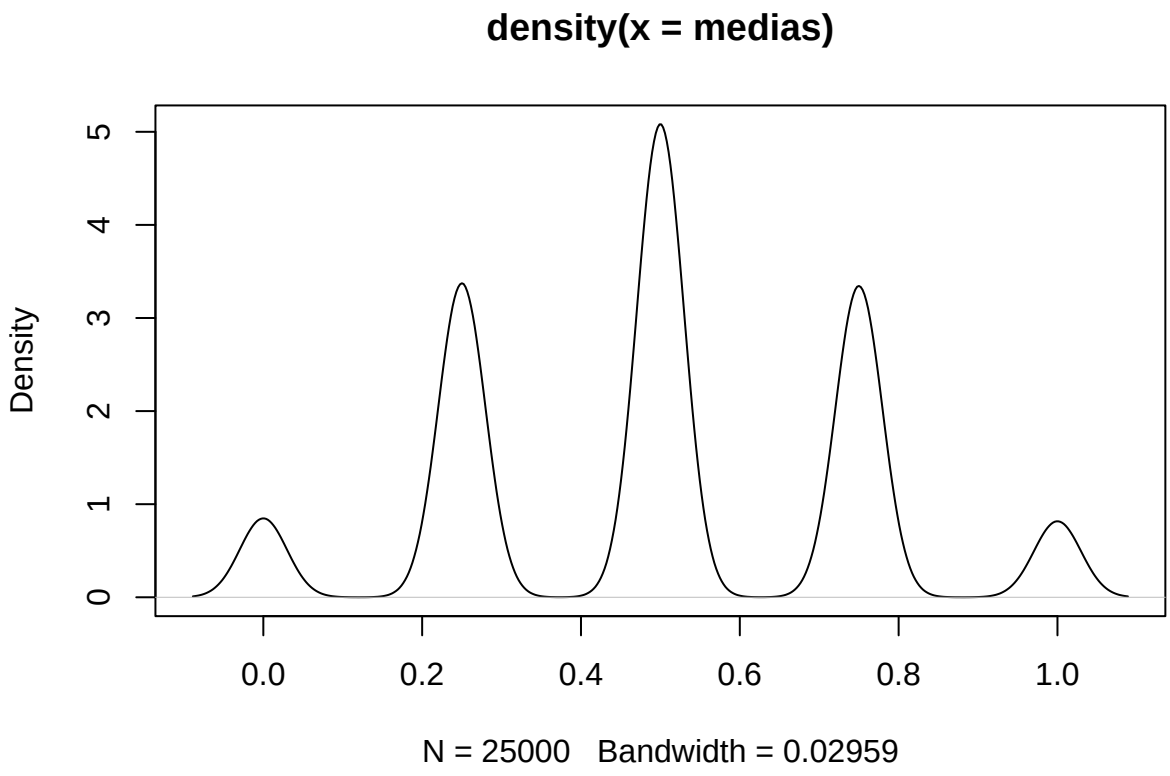
```
## [1] "Sd teórica: 0.08 -- Sd empírica: 0.08"
```

```
set.seed(3)

#con n<5

n2 <- 4

x <- seq(0,100,0.01)
```

```r
poblacion_n2 <- rbinom(x,1,0.5)

mu_pob <- mean(poblacion_n2)
sd_pob <- sd(poblacion_n2)

medias <- c()

for (i in 1:k){

  muestras <- sample(poblacion_n2,n2,replace = T)

  medias[i] <- mean(muestras)
}

hist(medias)
```




```r
plot(density(medias))
```




```
#Comprobacion teórica vs empírica

mu_teo <- mu_pob
sd_teo <- sd_pob/sqrt(n2)

mu_emp <- mean(medias)
sd_emp <- sd(medias)

sprintf("Media teórica: %.2f -- Media empírica: %.2f", mu_teo, mu_emp)
```

```
## [1] "Media teórica: 0.50 -- Media empírica: 0.50"
```

```
sprintf("Sd teórica: %.2f -- Sd empírica: %.2f", sd_teo, sd_emp)
```

```
## [1] "Sd teórica: 0.25 -- Sd empírica: 0.25"
```

Tarea 5

La distribución muestral de la varianza es más compleja ya que no sigue una distribución normal. Sin embargo, tiene algunas propiedades muy interesantes:

- La esperanza de la varianza es:

$$\mu_{S^2} = \frac{n-1}{n} * \sigma^2$$

- La esperanza de la varianza insesgada es:

$$S^2_{n-1} = \frac{\sum(X - \bar{X})^2}{n-1}$$

- La varianza de nS^2/σ^2 es 2(n-1)

$$\sigma^2_{nS^2/\sigma^2} = 2(n-1)$$

- La distribución de nS^2/σ^2 es una chi-cuadrado con n-1 grados de libertad:

$$nS^2/\sigma^2 \rightarrow \chi^2_{n-1}$$

Muestra con R las afirmaciones anteriores.

```
set.seed(3)


#Esperanza de la varianza

N=10000
n <- 30
sigma <- 4
mu <- 0

varianzas <- c()

for (i in 1:N){

  muestra <- rnorm(n,mu,sd=sqrt(sigma))

  varianzas[i] <- var(muestra)

}

esperanza_emp <- mean(varianzas)

esperanza_teo <- ((n-1)/n)* sigma

cat("Esperanza empírica de la varianza muestral:", esperanza_emp, "\n")
```

```
## Esperanza empírica de la varianza muestral: 4.007578
```

```
cat("Esperanza teórica de la varianza muestral:", esperanza_teo, "\n")
```

```
## Esperanza teórica de la varianza muestral: 3.866667
```

```r
# Esperanza varianza insesgada

# Calculamos la varianza insesgada (por defecto en R) de una muestra aleatoria
muestra <- rnorm(n, mean = mu, sd = sqrt(sigma))
varianza_insesgada <- var(muestra)

# Comparamos con la fórmula manual
varianza_manual <- sum((muestra - mean(muestra))^2) / (n-1)

cat("Varianza insesgada calculada con var():", varianza_insesgada, "\n")
```

```
## Varianza insesgada calculada con var(): 4.657741
```

```r
cat("Varianza insesgada calculada manualmente:", varianza_manual, "\n")
```

```
## Varianza insesgada calculada manualmente: 4.657741
```

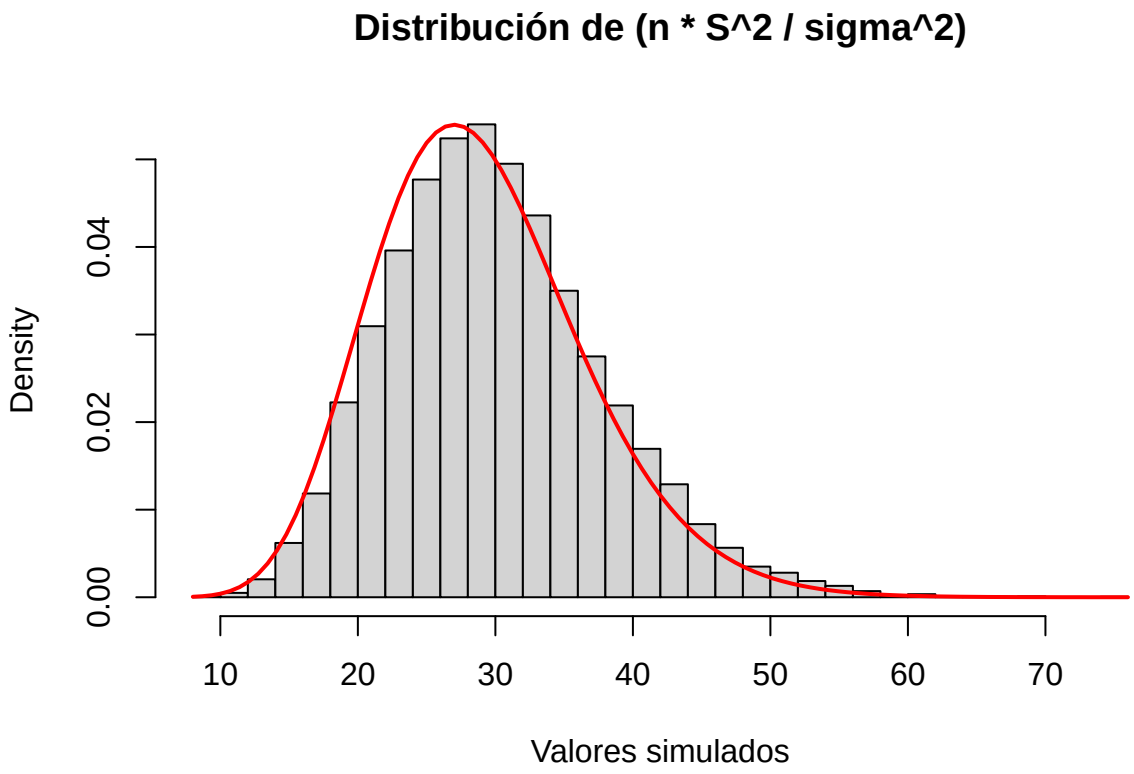
```r
#Varianza (n * S^2 / sigma^2)

# Calculamos la variable escalada
chi_squared_vals <- (n * varianzas) / sigma

# Calculamos la varianza de esta variable
var_chi_squared_empirica <- var(chi_squared_vals)

# Varianza teórica
var_chi_squared_teorica <- 2 * (n-1)

cat("Varianza empírica de (n * S^2 / sigma^2):", var_chi_squared_empirica, "\n")
```

```
## Varianza empírica de (n * S^2 / sigma^2): 61.90631
```

```r
cat("Varianza teórica de (n * S^2 / sigma^2):", var_chi_squared_teorica, "\n")
```

```
## Varianza teórica de (n * S^2 / sigma^2): 58
```

```r
#Comprobacion distribucion chi-cuadrado con n-1 grados de libertad

# Graficamos el histograma de los valores simulados
hist(chi_squared_vals, breaks = 30, probability = TRUE, main = "Distribución de (n * S^2 / sigma^2)",
     xlab = "Valores simulados")

# Superponemos la distribución teórica chi-cuadrado
curve(dchisq(x, df = n-1), col = "red", lwd = 2, add = TRUE)
```



Los intervalos de confianza

Si la distribución de la media es como hemos dicho $N(\mu, \frac{\sigma}{\sqrt(n)}$ podemos construir un intervalo de confianza siguiendo la regla:

$$IC : \mu \pm Z_{\alpha/2} * \frac{\sigma}{\sqrt(n)}$$

Donde $Z_{\alpha/2}$ son los valores de 1.96 y 2.58 para valores de alfa de 0.05 y 0.01 respectivamente.

Tarea 6

Un intervalo construido con la fórmula anterior, cumple, por lógica, la siguiente afirmación:

Con un α=0.05 al 95% de las muestras extraías de una población tendrán una media que caerá dentro de los límites del itnervalo, y con un α=0.01 caerán al 99% de las medias.

Compruebe con R la afirmación anterior.

Recuerde que estamos calculando los intervalos utilizando la distribución normal, porque en nuestras simulaciones creamos nosotros las poblaciones, y por tanto μ y σ (los parámetros poblacionales) son conocdios. En general, con poblaciones reales, estos parámetros son desconocidos por lo que sustituimos $Z_{\alpha/2}$ por $t_{n-1,\alpha/2}$. En este caso los itnervalos de confianza para la media se calculan de la siguiente manera:

$$IC : \bar{X} \pm t_{n-1,\alpha/2} * \frac{S_x}{\sqrt(n)}$$

A efectos prácticos, ambas distribuciones son tan similares con muestras mayores de 30 (lo estudiamos ya) que podemos simplificar y usar los valores convencionales de 1.96 y 2.58 para valores de alfa de 0.05 y 0.01 respectivamente.

```
set.seed(3)


n <- 1000

poblacion <- rnorm(n,0,1)

m_poblacion <- mean(poblacion)
sd_poblacion <- sd(poblacion)

#Intervalos teoricos
#0.95

sup_poblacion95 <- m_poblacion+1.96*(sd_poblacion/sqrt(n))
inf_poblacion95 <- m_poblacion-1.96*(sd_poblacion/sqrt(n))

#0.99

sup_poblacion99 <- m_poblacion+2.58*(sd_poblacion/sqrt(n))
inf_poblacion99 <- m_poblacion-2.58*(sd_poblacion/sqrt(n))



calcular_ic <- function(datos,media,desviacion,alfa){

  z <- qnorm(1-alfa/2)

  error_tipico <- z*desviacion/sqrt(length(datos))

  inf <- media-error_tipico
  sup <- media+error_tipico

  return(c(inf,sup))
}


muestras <- matrix(0, nrow=25000, ncol=length(poblacion))
medias <- numeric(25000)
desviaciones <- numeric(25000)


for (i in 1:25000){

  muestras[i,] <- rnorm(length(poblacion), 0, 1)

  medias[i] <- mean(muestras[i,])

  desviaciones[i] <- sd(muestras[i,])}
```

```r
intervalos99 <- matrix(0,25000,3)

for (i in 1:25000){

  intervalos99[i,1] <- calcular_ic(poblacion, medias[i],desviaciones[i],0.01)[1]

  intervalos99[i,2] <- calcular_ic(poblacion,medias[i], desviaciones[i], 0.01)[2]

  intervalos99[i,3] <- ifelse(intervalos99[i,1]>inf_poblacion99 | intervalos99[i,2]<sup_poblacion99,1,0)

}

intervalos95 <- matrix(0,25000,3)

for (i in 1:25000){

  intervalos95[i,1] <- calcular_ic(poblacion,medias[i],desviaciones[i], 0.05)[1]
  intervalos95[i,2] <- calcular_ic(poblacion,medias[i],desviaciones[i],0.05)[2]

  intervalos95[i,3] <- ifelse(intervalos95[i,1]>inf_poblacion95 | intervalos95[i,2]<sup_poblacion95,1,0)

}

table(intervalos95[,3])
```

```
##
##     0     1
##   370 24630
```

```r
table(intervalos99[,3])
```

```
##
##     0     1
##   456 24544
```

```r
porcentaje95 <- mean(intervalos95[,3]==1)*100
porcentaje99 <- mean(intervalos99[,3]==1)*100

cat("Con alfa = 0.05, el", porcentaje95, "% de las medias caen dentro del intervalo.\n")
```

```
## Con alfa = 0.05, el 98.52 % de las medias caen dentro del intervalo.
```

```r
cat("Con alfa = 0.01, el", porcentaje99, "% de las medias caen dentro del intervalo.\n")
```

```
## Con alfa = 0.01, el 98.176 % de las medias caen dentro del intervalo.
```