Practica 16

Cristian Zaldaña

2022-09-06

Índice

TEOREMA DEL LÍMITE CENTRAL

El Teorema del Límite Central (TLC) informa acerca de la distribución de muestreo de medias de muestras con tamaño n . Recuérdese que básicamente existen tres tipos de información que se desea conocer sobre una distribución:

- dónde está el centro,
- qué tanto varía, y
- cómo está repartida.

El Teorema del Límite Central establece que sí las observaciones $X_1,\ X_2\ ,X_3\ ,...,\ X_n$ son variables aleatorias independientes e idénticamente distribuidas con una distribución de probabilidad cualquiera y en la cual cada una de ellas tenga la misma media μ y la misma varianza σ^2 (ambas finitas).

Entonces el promedio muestral tiene una distribución con media μ y varianza $\frac{\sigma^2}{n}$que tiende hacia una distribución $N(0,1)$ a medida que n tiende a ∞.

¿Cómo podemos comprobar esto?

La simulación es un excelente camino.

Simular el Teorema del Límite Central con datos binomial

Consideremos n repeticiones independientes y sea X el número de veces que ocurre un suceso A. Sea p igual a $P(A)$ y supongamos que este número es constante para todas las repeticiones consideradas.

El teorema central del límite nos indica que:

$\frac{X-E[X]}{\sqrt{V(X)}} = \frac{X-np}{\sqrt{npq}}$ es aproximadamente $N(0,1)$

- **Ejemplo 1:** Generar 100 números aleatorios de una distribución binomial con parámetros n=10 (número de ensayos o pruebas), y p=0.25 (probabilidad de éxito)

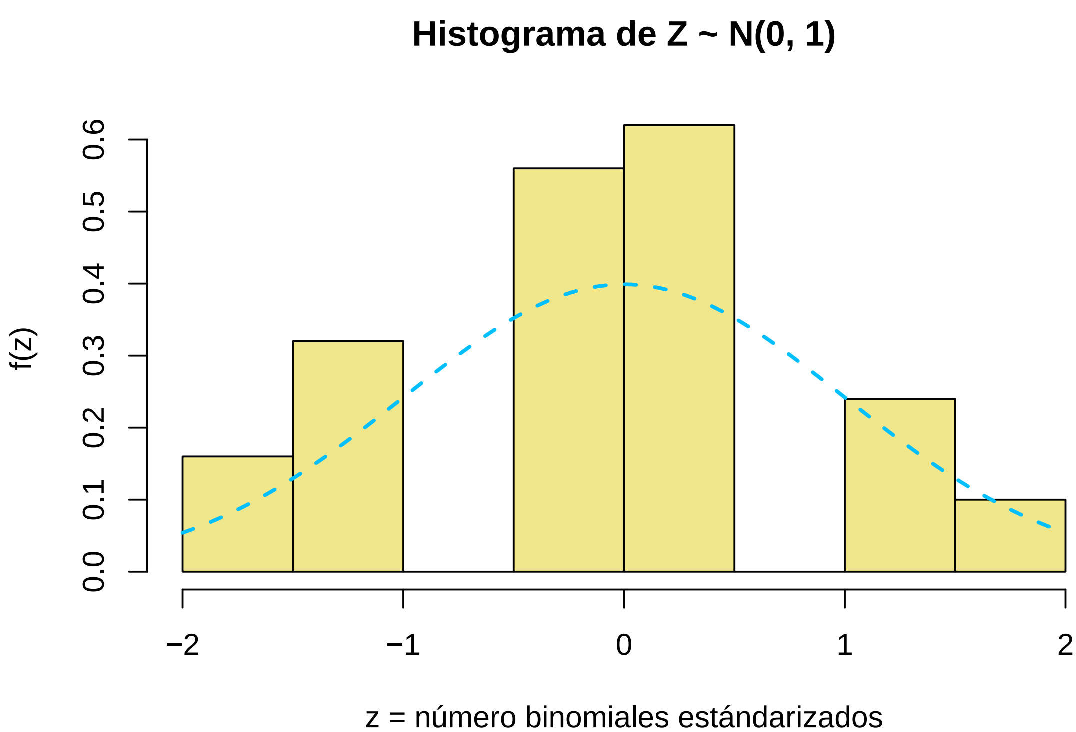
```
# tm= tamaño de la muestra
tm=100
n <- 10
p <- 0.25
#generando las 100 números aleatorios
S = rbinom(tm, n, p)
# estandarizando cada una de las observaciones
Z = (S-n*p)/sqrt(n*p*(1-p))
Z
```

```
##   [1]  0.3651484 -0.3651484  0.3651484 -1.0954451  1.0954451 -1.8257419
##   [7] -0.3651484  0.3651484  0.3651484  0.3651484  0.3651484  1.0954451
##  [13]  0.3651484  0.3651484 -1.0954451 -0.3651484 -0.3651484  1.8257419
##  [19] -1.0954451 -1.8257419  1.0954451 -0.3651484  1.8257419 -1.0954451
##  [25] -0.3651484  0.3651484  1.8257419 -1.8257419 -1.8257419  0.3651484
##  [31] -0.3651484 -1.0954451  1.0954451  0.3651484 -0.3651484 -0.3651484
##  [37]  0.3651484  0.3651484 -0.3651484 -1.0954451  1.0954451 -0.3651484
##  [43]  0.3651484  0.3651484 -1.0954451  1.0954451  0.3651484  1.0954451
##  [49]  0.3651484  1.0954451  1.8257419 -1.8257419 -1.0954451 -0.3651484
##  [55] -0.3651484  0.3651484  0.3651484  1.0954451 -1.0954451 -1.8257419
##  [61] -0.3651484  0.3651484  0.3651484  0.3651484  0.3651484  0.3651484
##  [67] -1.8257419 -0.3651484  1.8257419 -0.3651484  0.3651484  0.3651484
##  [73] -0.3651484 -0.3651484 -1.0954451  0.3651484  0.3651484 -1.0954451
```

```
## [79] -0.3651484 -0.3651484  1.0954451 -0.3651484 -0.3651484 -1.0954451
## [85] -0.3651484  1.0954451 -1.0954451 -1.0954451 -1.8257419 -1.0954451
## [91]  1.0954451 -0.3651484  0.3651484 -0.3651484  0.3651484  0.3651484
## [97] -0.3651484 -1.0954451 -0.3651484 -0.3651484
```

La variable X tiene los resultados, y podemos ver la distribución de los números aleatorios en X con un histograma

```
hist(Z, main="Histograma de Z ~ N(0, 1)", xlab="z = número binomiales estándarizados",
ylab="f(z)", prob=TRUE, col="khaki")
curve(dnorm(x, 0, 1), col = "deepskyblue", lty=2, lwd=2, add=TRUE)
```



La distribución muestra un gráfico aproximadamente normal. Esto es, en forma de campana, centrada en 0 y con desviación estándar 1.

- Simular el TLC con datos de una distribución normal.

El teorema central del límite establece que $\frac{\bar{X}-\mu}{\sigma/\sqrt{n}} \sim N(0,1)$

- **Ejemplo 2:**

Suponga que X_i es normal con media $\mu = 5$ y desviación estándar $\sigma = 5$. Entonces necesitamos una función para encontrar el valor de $\frac{\bar{X}-\mu}{\sigma/\sqrt{n}}$

```
simulNorm<-function(mu,sigma,m=5,n=100)
{
  vectMedias<<-numeric(0)
  MediasEstand<<-numeric(0)
  for(i in 1:m)
```

```
  {
    X = rnorm(n,mu,sigma)
    # genera n valores normales
    vectMedias[i]<<-mean(X)
    MediasEstand[i]<<-(vectMedias[i]-mu)/(sigma/sqrt(n))
  }
}

mu=5
sigma=5
m<-200
# numero de muestras o medias a obtener

simulNorm(mu,sigma,m)

hist(MediasEstand, main="Histograma de medias estandarizadas",
     xlab="Valores de m medias normales estandarizadas",prob=TRUE, col="darkolivegreen3")
curve(dnorm(x,0,1),col="deeppink3",lty=2,lwd=2,add=TRUE)
```

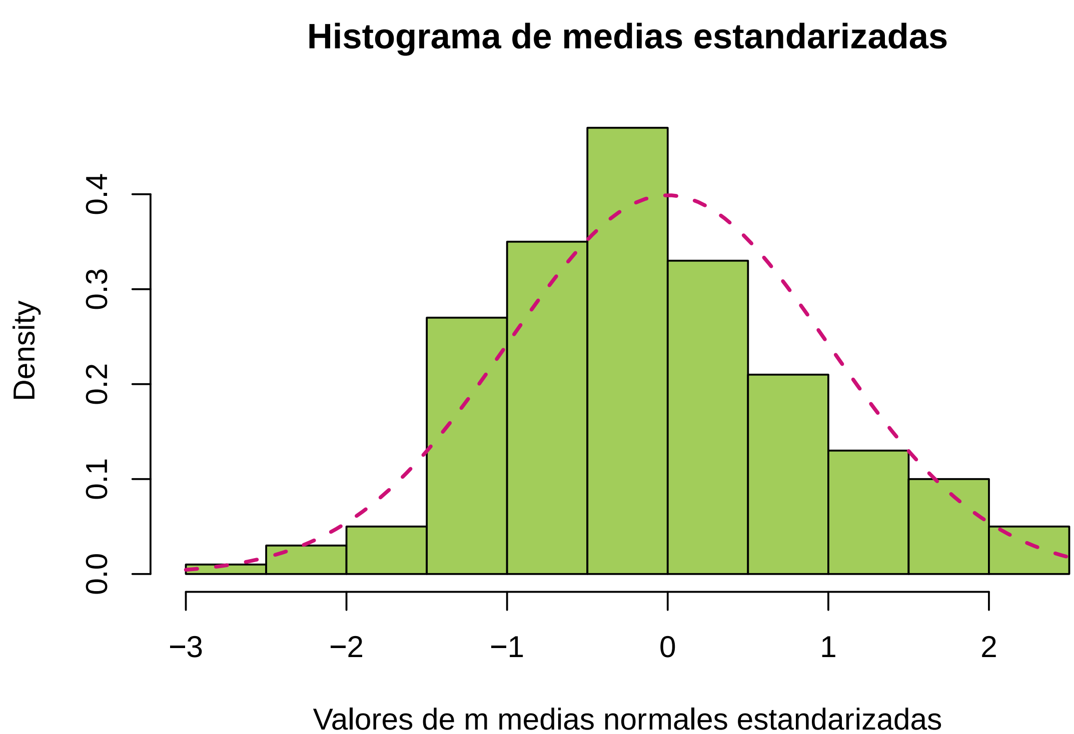


Gráfico de “probabilidad normal”

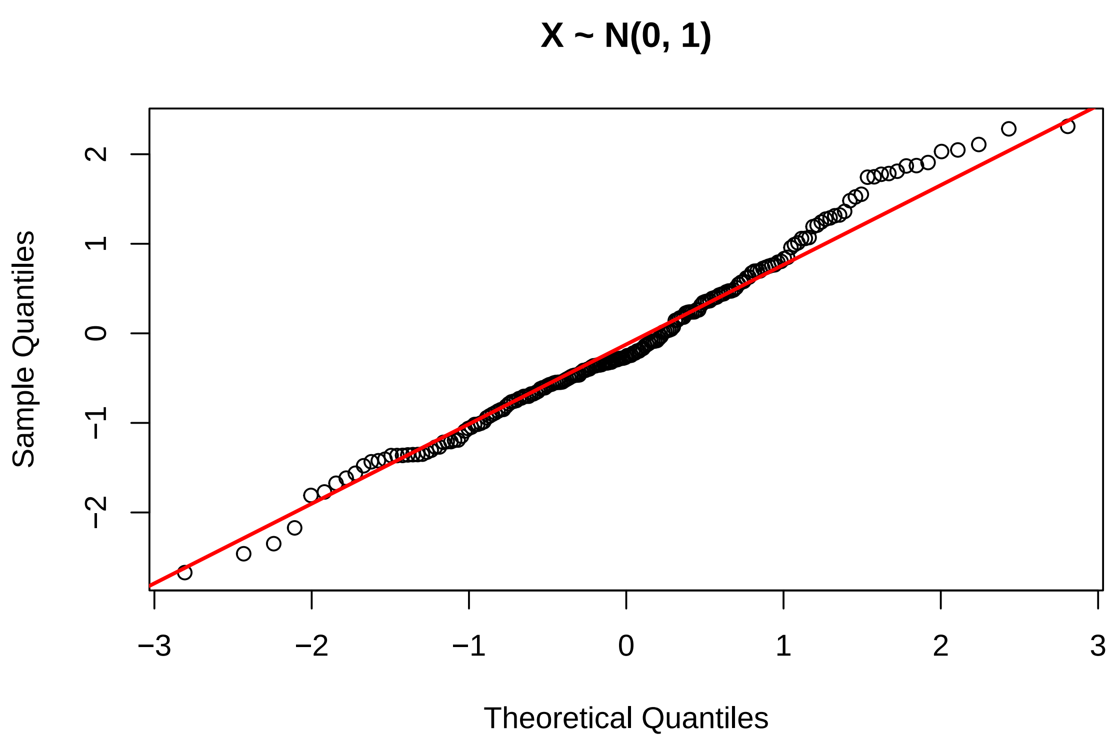
Un mejor gráfico que el histograma para decidir si los datos aleatorios son aproximadamente normal es el llamado gráfico de “probabilidad normal”. La idea básica es graficar los cuantiles de sus datos contra los correspondientes cuantiles de la distribución normal. Los cuantiles de un conjunto de datos preferidos son la Mediana, Q_1 y Q_3 los más generales. El cuantil q es el valor en los datos donde q*100%.

También el cuantil 0.25 es Q_1, el cuantil 0.5 es la mediana y el cuantil 0.75 es Q_3. Los cuantiles para la distribución teórica son similares, sólo cambia el número de puntos datos menores, o sea el área a la izquierda del monto especificado. Por ejemplo, la mediana parte el área por debajo de la curva de densidad en la mitad.\

El gráfico de probabilidad normal es fácil de leer si conoce cómo. Esencialmente, si el gráfico parece una línea recta entonces los datos son aproximadamente normal. Está línea no es una línea de regresión. La línea es trazada a través de los puntos formados por el primer y tercer cuartil.\

R hace todo esto fácil con las funciones qqnorm(), más generalmente qqplot(), y qqline() la cual traza una línea de referencia (no una línea de regresión).

```
qqnorm(MediasEstand, main="X ~ N(0, 1)")
#muestra la línea
qqline(MediasEstand, lty=1, lwd=2, col="red")
```



Simular el Teorema del Límite Central con datos exponencial

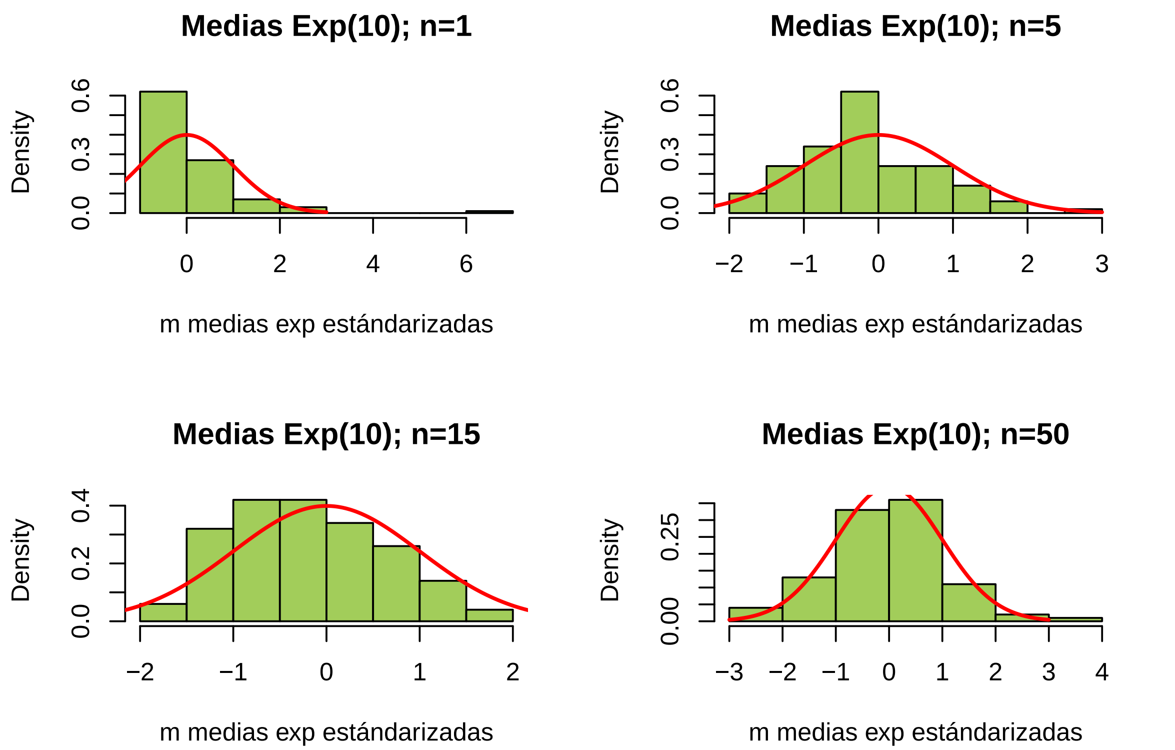
Un ejemplo de una distribución sesgada es la exponencial. Necesitamos conocer que sí tiene media $\mu = 10$, entonces la desviación estándar σ es también 10, por eso sólo necesitamos especificar la media.\

Vamos a simular para varios valores de n . Para cada una de las m =100 muestras, n será 1, 5, 15, 50 (el número de valores aleatorios en cada uno de los promedios).

```
simulExp <- function(mu, m=5, n=100)
{
 razon <- 1/mu
 vectMedias <<- numeric(0)
```

```
 MediasEstand <<- numeric(0)
 for (i in 1:m)
{
 X = rexp(n, razon)
# genera n valores exponenciales
 vectMedias[i] <<- mean(X)
 MediasEstand[i] <<- (vectMedias[i] - mu)/(mu/sqrt(n))
 }
 }

par(mfrow=c(2,2))
# para n=1
 mu=10
 m <- 100; n <- 1
 simulExp(mu, m, n)
 hist(MediasEstand, main="Medias Exp(10); n=1", xlab="m medias exp estándarizadas",
prob=TRUE, col="darkolivegreen3")
 xvals = seq(from=-3, to=3, by=0.01)
 points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
# para n=5
 n <- 5
 simulExp(mu, m, n)
 hist(MediasEstand, main="Medias Exp(10); n=5", xlab="m medias exp estándarizadas",
prob=TRUE, col="darkolivegreen3")
 xvals = seq(from=-3, to=3, by=0.01)
 points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
 # Repita este proceso para n=15 y n=50
 # para n=15
 n <- 15
 simulExp(mu, m, n)
 hist(MediasEstand, main="Medias Exp(10); n=15", xlab="m medias exp estándarizadas",
prob=TRUE, col="darkolivegreen3")
 xvals = seq(from=-3, to=3, by=0.01)
 points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
 # para n=5
 n <-50
 simulExp(mu, m, n)
 hist(MediasEstand, main="Medias Exp(10); n=50", xlab="m medias exp estándarizadas",
prob=TRUE, col="darkolivegreen3")
 xvals = seq(from=-3, to=3, by=0.01)
 points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
```



Observe que el histograma tiene una forma muy acampanada entre n=15 y n=50, aunque justo en n=50 parece todavía ser un poco sesgada.\

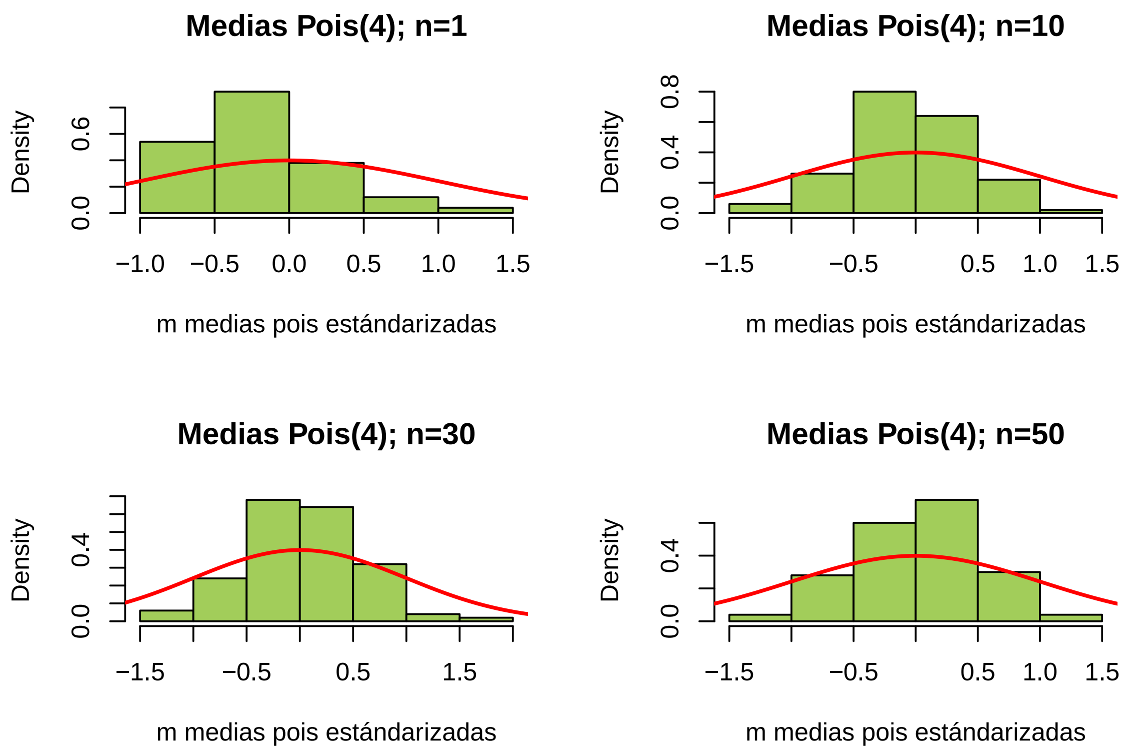
- **Ejercicios.**

Simular el Teorema del Límite Central para una variable aleatoria que tiene distribución Poisson con lambda o media 4. Considerar 100 muestras aleatorias de tamaño 1, 10, 30, 50 valores de la distribución. Los gráficos deben estar en una misma ventana.

```
simulPois <- function(mu, m=5, n=100)
{
 vectMedias <<- numeric(0)
 MediasEstand <<- numeric(0)
 for (i in 1:m)
{
 X = rpois(n,mu)
# genera n valores de la distribución de Poisson.
 vectMedias[i] <<- mean(X)
 MediasEstand[i] <<- (vectMedias[i] - mu)/(mu/sqrt(n))
 }
 }

par(mfrow=c(2,2))
# para n=1
 mu=4
 m <- 100; n <- 1
 simulPois(mu, m, n)
```

```r
hist(MediasEstand, main="Medias Pois(4); n=1", xlab="m medias pois estándarizadas",
prob=TRUE, col="darkolivegreen3")
xvals = seq(from=-3, to=3, by=0.01)
points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
# para n=10
mu=4
m <- 100; n <- 10
simulPois(mu, m, n)
hist(MediasEstand, main="Medias Pois(4); n=10", xlab="m medias pois estándarizadas",
prob=TRUE, col="darkolivegreen3")
xvals = seq(from=-3, to=3, by=0.01)
points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
# para n=1
mu=4
m <- 100; n <- 30
simulPois(mu, m, n)
hist(MediasEstand, main="Medias Pois(4); n=30", xlab="m medias pois estándarizadas",
prob=TRUE, col="darkolivegreen3")
xvals = seq(from=-3, to=3, by=0.01)
points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
# para n=1
mu=4
m <- 100; n <- 50
simulPois(mu, m, n)
hist(MediasEstand, main="Medias Pois(4); n=50", xlab="m medias pois estándarizadas",
prob=TRUE, col="darkolivegreen3")
xvals = seq(from=-3, to=3, by=0.01)
points(xvals, dnorm(xvals, 0, 1), col = "red", type="l", lty=1, lwd=2)
```


Medias Pois(4); n=1
Density
0.0
0.6
−1.0
−0.5
0.0
0.5
1.0
1.5
m medias pois estándarizadas
Medias Pois(4); n=10
Density
0.0
0.4
0.8
−1.5
−0.5
0.5
1.0
1.5
m medias pois estándarizadas
Medias Pois(4); n=30
Density
0.0
0.4
−1.5
−0.5
0.5
1.5
m medias pois estándarizadas
Medias Pois(4); n=50
Density
0.0
0.4
−1.5
−0.5
0.5
1.0
1.5
m medias pois estándarizadas