Binomial 2 Normal

J. Abellán

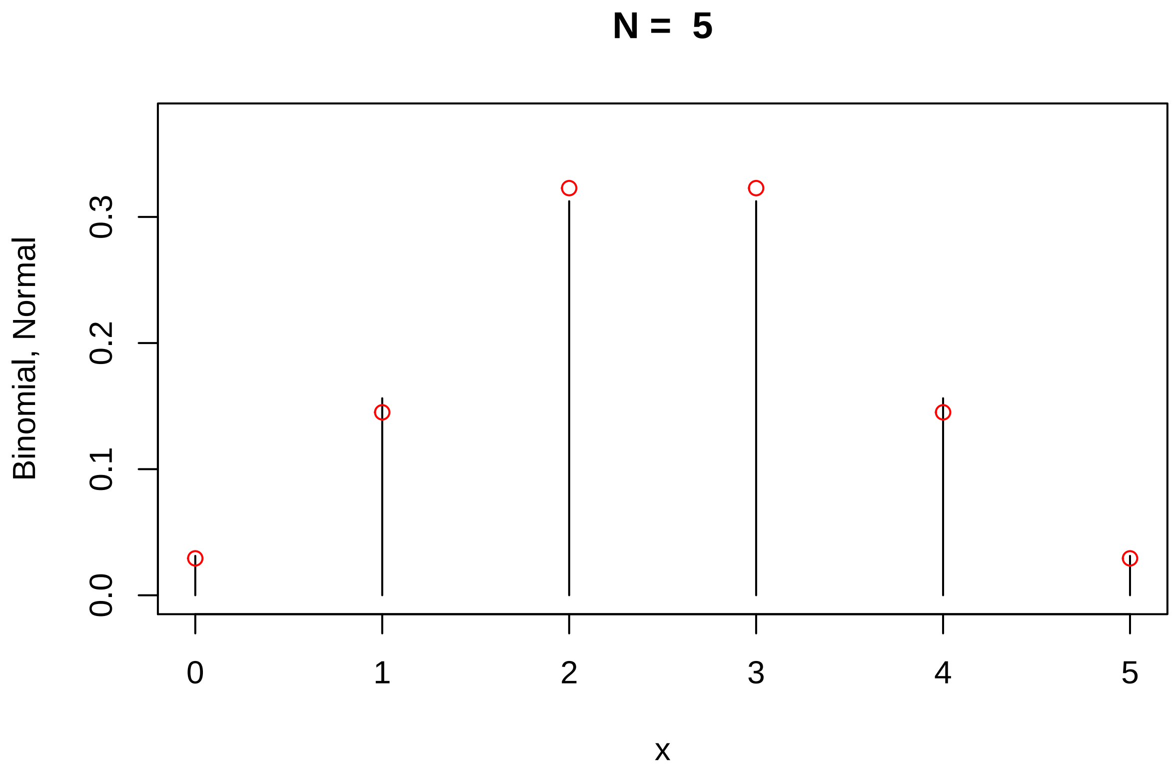
22/10/2017

La normal como límite de la binomial

El objetivo de esta práctica es comprobar la bondad de la aproximación de la binomial de parámetros N, θ por una gaussiana de parámetros $\mu = N\theta$ y $\sigma = \sqrt{N\theta(1-\theta)}$ conforme $N \to \infty$.

No olviden en ningún momento que la binomial es una función de distribución **discreta** y la normal es **contínua**.

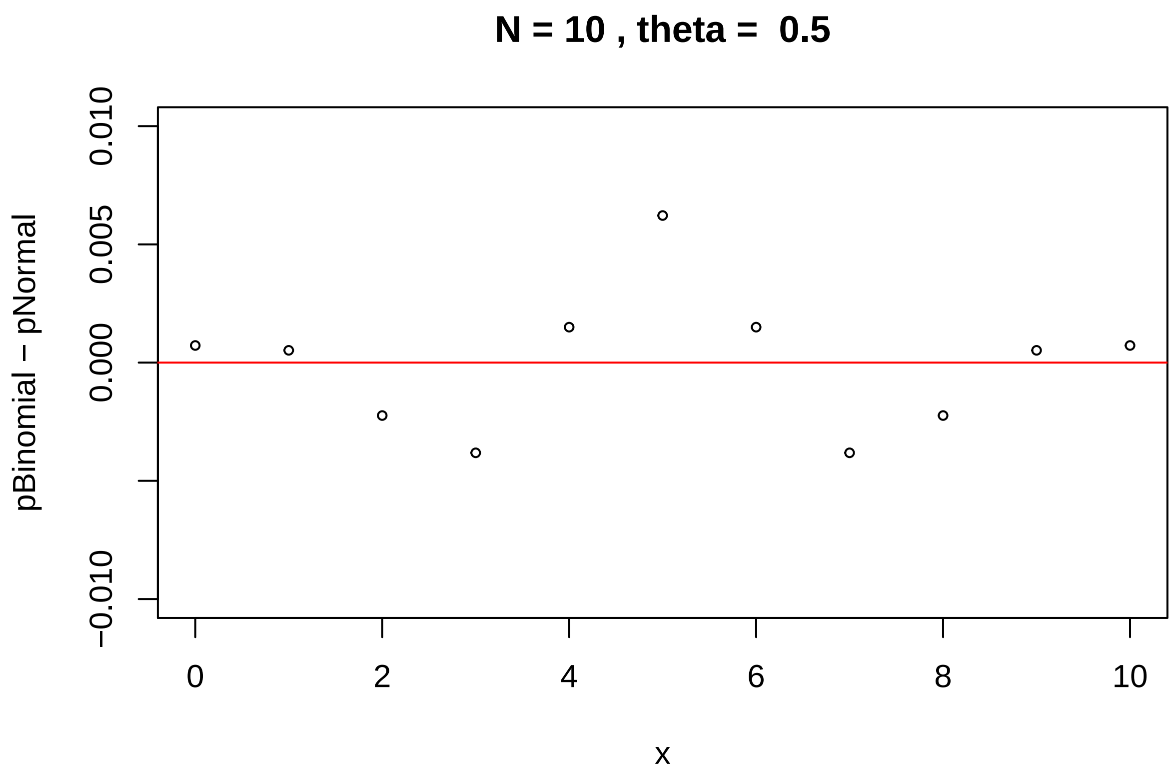
```r
#library("latex2exp", lib.loc="~/R/i686-pc-linux-gnu-library/3.2")

#Binomial de parámetros N, theta
N <- 5 ; theta <- 0.5

#Valores posibles de la variable aleatoria
x <- 0 : N

py <- dbinom( x, N, theta )

plot( x, py,

      type = "h",

      ylim = c( 0, 1.2 * max( py ) ),

      ylab = "Binomial, Normal",

      main = paste( "N = ", N )

    )

#Aproximación mediante una gaussiana de igual media y varianza
mu <- N * theta

varx <- N * theta * ( 1 - theta )

sigma <- sqrt( varx )

# La normal
#pya = (1 / sqrt(2pi) sigma ) exp(- 0.5 ( x - mu )^2 / sigma^2 )
pya <- dnorm( x, mu, sigma )

points( x, pya, col = 2 )
```

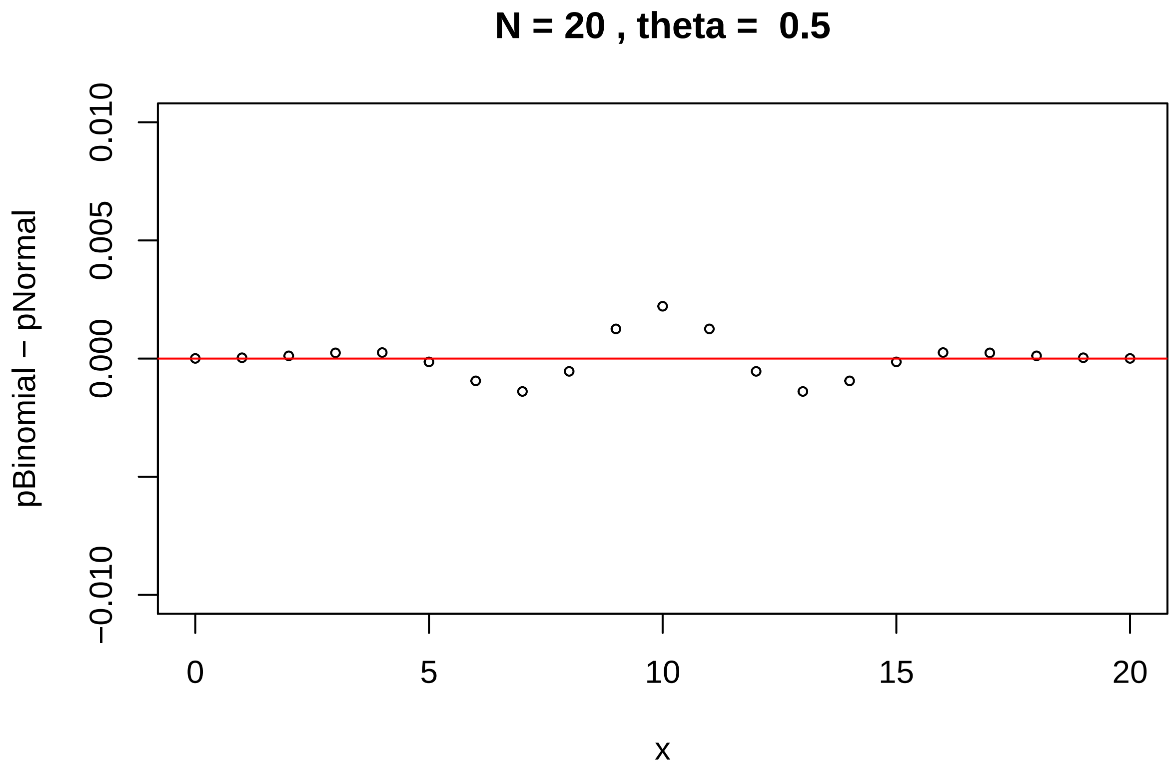


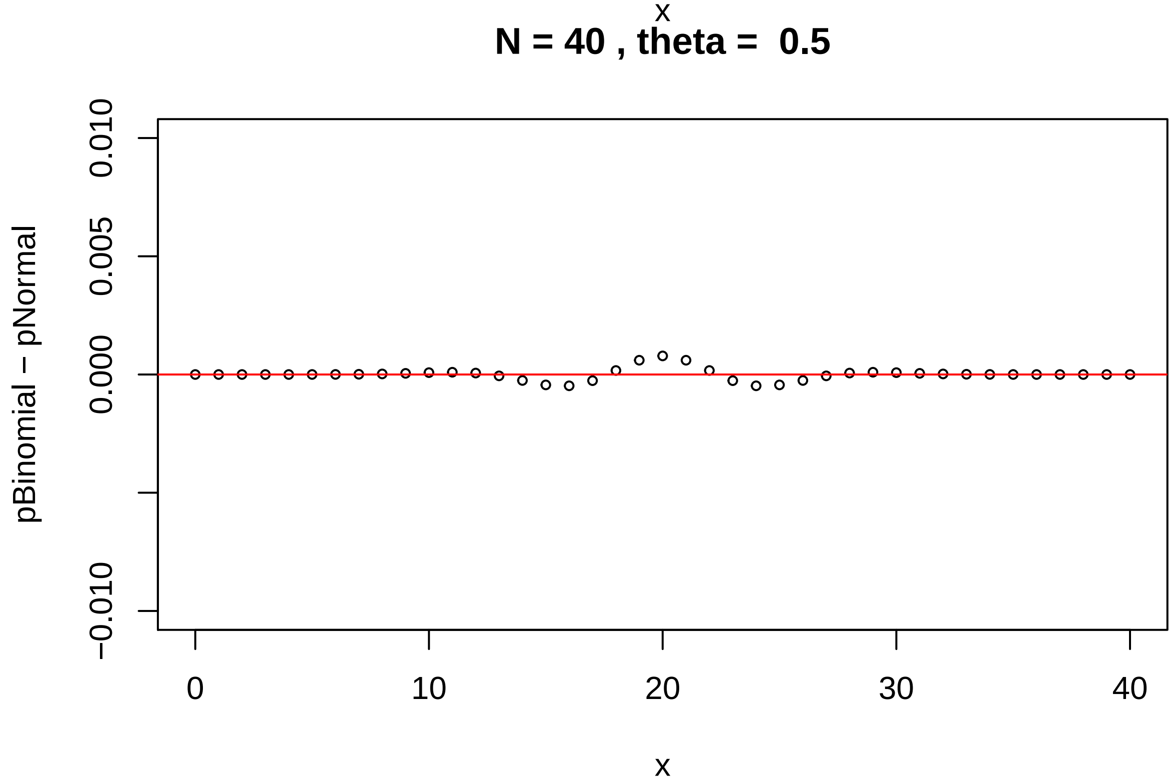
Se ve mejor la bondad de la aproximación si dibujamos las diferencias conforme aumenta el parámetro N:

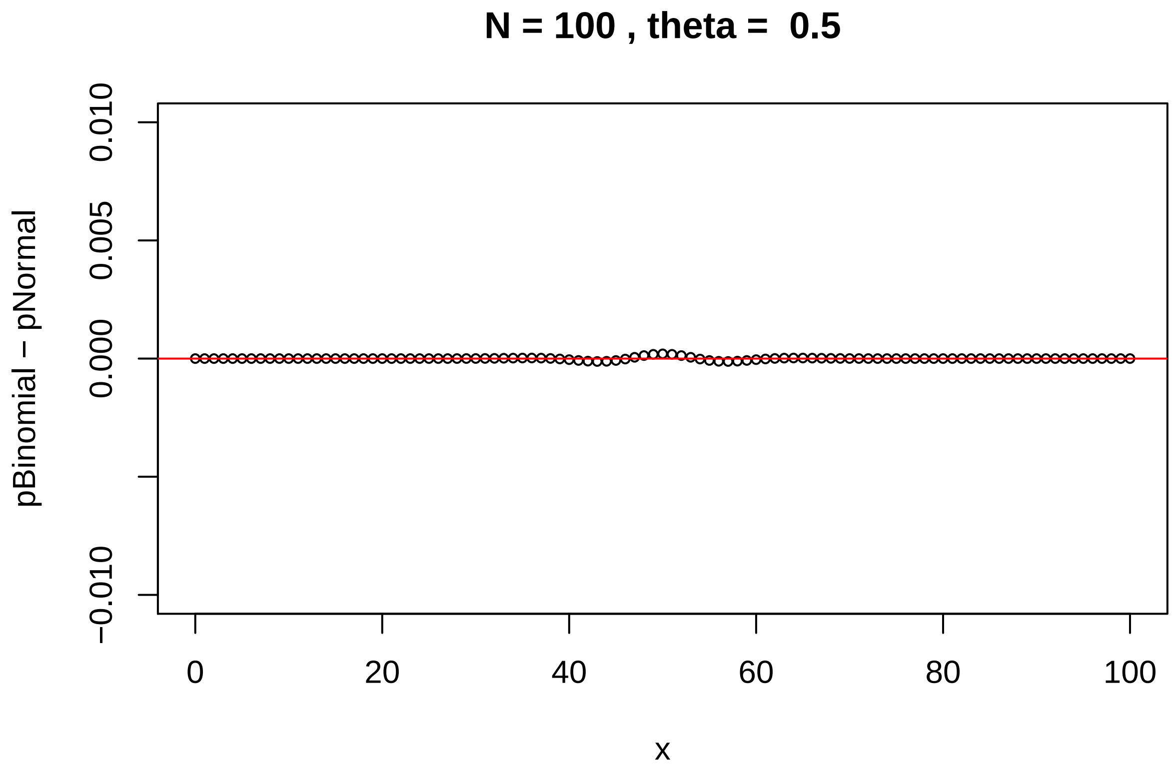
```
#Binomial de parámetros N, theta
theta <- 0.5

N <- c( 10, 20, 40, 100 )

nN <- length( N )

#Escala de dibujo, error máximo
ey <- 1e-2

for (i in seq( along = N ) ) {

  # número de lanzamientos y parámetros asociados
  n <- N[ i ]

  mu <- n * theta

  varx <- n * theta * ( 1 - theta )

  sigma <- sqrt( varx )

  #Valores posibles de la variable aleatoria
  x <- 0 : n

  # La binomial
  py <- dbinom( x, n, theta )

  # La normal:
```

```r
  # ya = (1 / sqrt(2pi) sigma) exp(- 0.5 ( x - mu)^2 / varx )
  pya <- dnorm( x, mu, sigma)

  plot( x, pya - py,

        cex = .6,

        ylim = c( - ey, ey ),

        ylab = "pBinomial - pNormal",

        main = paste("N =", n, ", theta = ", theta )

      )

  abline( h = 0, col = 2 )

}
```




N = 20 , theta = 0.5
pBinomial − pNormal
0.010
0.005
0.000
−0.010
0
5
10
15
20
x

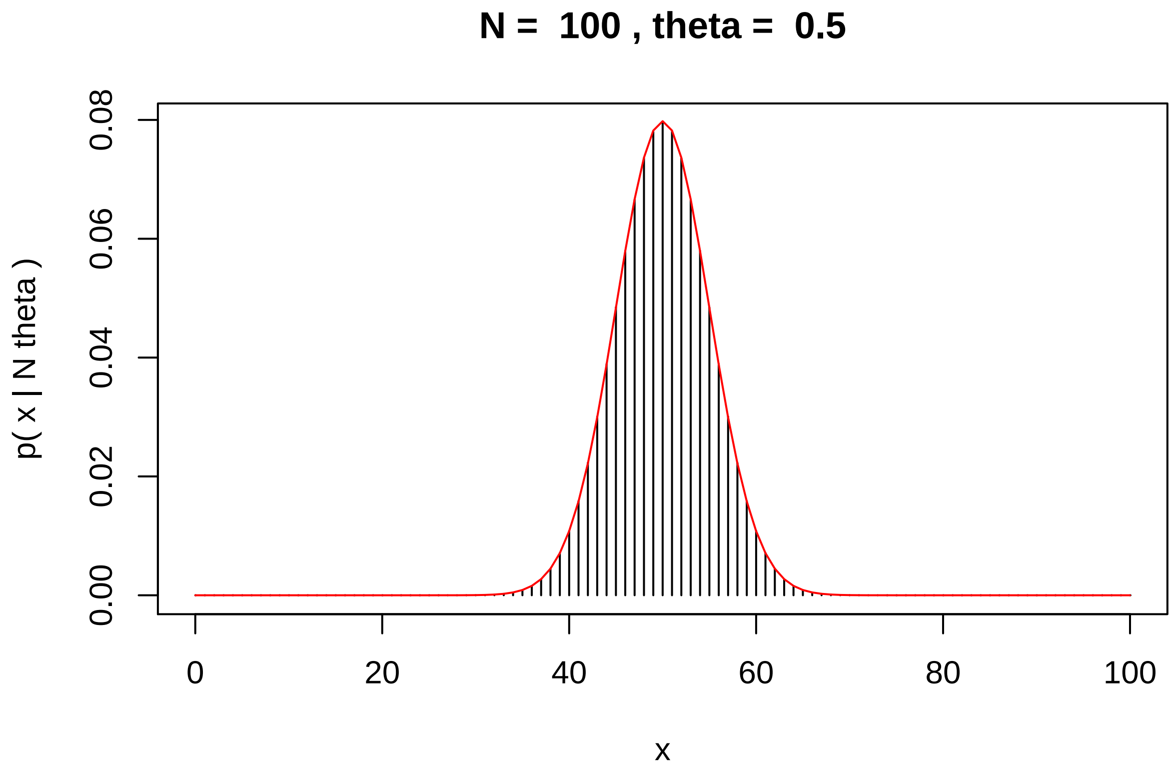

N = 40 , theta = 0.5
pBinomial − pNormal
0.010
0.005
0.000
−0.010
0
10
20
30
40
x



El resultado final:

Superpongo a la binomial discreta (en negro) la normal en rojo.

```
plot( x, py,

      type = "h",

      #ylab = latex2exp("$p( x | N \\theta )$"),
      ylab = "p( x | N theta )",

      main = paste( "N = ", n, ", theta = ", theta )

    )

lines( x, pya, col = 2 )
```


N = 100 , theta = 0.5
p(x | N theta)
0.00
0.02
0.04
0.06
0.08
0
20
40
60
80
100
x