Tarea 2

Rolando Corona Jiménez

11 de septiembre de 2018

Contents

Ejercicio 1

Una pareja decide tener hijos hasta el nacimiento de la primera niña. Calcule la probabilidad de que tengan más de 4 hijos. Suponga que las probabilidades de tener niño o niña son iguales. ¿Cuál es el tamaño esperado de la familia?

Si X denota el número de hijos nacidos hasta que nace la primera niña, entonces $X \sim Geom(\frac{1}{2})$, pues se asume que la probabilidad de tener niño o niña es la misma. Con esto en cuenta, la probabilidad de que tengan más de cuatro hijos es $P(X > 4)$, que se calcula como

$$\begin{aligned}
P(X > 4) &= 1 - P(X \leq 4) && (1)\\
&= 1 - \sum_{i=1}^{4} (\frac{1}{2})^{i-1}(\frac{1}{2}) && (2)\\
&= 1 - \frac{1}{2}\Big(\frac{1-(\frac{1}{2})^4}{\frac{1}{2}}\Big) && (3)\\
&= 1 - (1 - (\frac{1}{2})^4) && (4)\\
&= (\frac{1}{2})^4 && (5)\\
&= 0.0625 && (6)
\end{aligned}$$

$$\mathbf{P(X > 4) = 0.0625}$$

El tamaño esperado de la familia es $\frac{1}{p}$, en este caso es $p = \frac{1}{2}$, por lo que

$$\mathbf{E(X) = 2}$$

Ejercicio 2

Cuando una máquina no se ajusta adecuadamente tiene una probabilidad 0.15 de producir un artículo defectuoso. Diariamente, la máquina trabaja hasta que se producen 3 artículos defectuosos. Se detiene la máquina y se revisa para ajustarla. ¿Cuál es la probabilidad de que una máquina mal ajustada produzca 5 o más artículos antes de que sea detenida? ¿Cuál es el número promedio de artículos que la máquina producirá antes de ser detenida?

Sea X el número de artículos producidos hasta que se producen 3 artículos defectuosos. Asumiendo independencia entre cada aparición de un artículo defectuoso, se tiene que $X \sim BN(3, .15)$.

Se desea calcular $P(X > 5)$, que se calcula como

$$\begin{aligned}
P(X > 5) &= 1 - P(X \leq 5) && (7)\\
&= 1 - (P(X = 3) + P(X = 4) + P(X = 3)) && (8)\\
&= 1 - (.003375 + 0.00860625 + 0.01463062) && (9)\\
&= 0.9733881 && (10)
\end{aligned}$$

donde $P(X = x) = \binom{x-1}{r-1} p^r (1-p)^{x-r}$

$$\mathbf{P(X > 5) = 0.9733881}$$

El número promedio de artículos que la máquina producirá antes de ser detenida es

$$\mathbf{E(X) = \frac{r}{p} = \frac{3}{.15} = 20}$$

Ejercicio 3

Los empleados de una compañía de aislantes son sometidos a pruebas para detectar residuos de asbesto en sus pulmones. Se le ha pedido a la compañía que envíe a tres empleados cuyas pruebas resulten positivas, a un centro médico para realizarles más análisis. Si se sospecha que el 40% de los empleados tienen residuos de asbesto en sus pulmones, encuentre la probabilidad de que deban ser analizados 10 trabajadores para poder encontrar a 3 con resultado positivo.

Si X denota el número de trabajadores analizados necesarios para que 3 de ellos muesten resultado positivo, entonces $X \sim BN(3, .4)$, ya que la probabildiad de que un tabajador dé resultado positivo es .4.

En este caso se desea conocer $P(X = 10)$, que es igual a

$$\binom{9}{2}(.4)^3(.6)^7 = 0.06449725$$

Por lo que

$$\mathbf{P(X = 10) = 0.06449725}$$

Ejercicio 4

(a) Considere una moneda desequilibrada que tiene probabilidad p de obtener águila. Usando el comando sample, escriba una función que simule N veces lanzamientos de esta moneda hasta obtener un águila. La función deberá recibir como parámetros a la probabilidad p de obtener águila y al número N de veces que se repite el experimento; y tendrá que regresar un vector de longitud N que contenga el número de lanzamientos hasta obtener un águila en cada uno de los N experimentos.

(b) Usando la función anterior simule $N = 10^4$ veces una variable aleatoria $Geom(p)$ para $p = 0.5; 0.1; 0.01$ Grafíque las frecuencias normalizadas en color azul. Sobre está última figura empalme en rojo la gráfica de la función de masa correspondiente. ¿Qué observa?

(c) Repita el inciso anterior para $N = 10^6$. Además calcule el promedio y la desviación estándar de las simulaciones que realizó ¿Qué observa?

Después de realizar múltiples simulaciones con $N = 10^4, 10^6$, se observa que las simulaciones son una buena aproximación a la distribución geométrica. Cabe observar que algunas gráficas puedan sugerir lo contrario, pero esto es debido principalmente a una cuestión de escalas.

A continuación se presenta el código desarrolado en R para este ejercicio, la función simula_lanzamientos(p,N) regresa un vector con las características deseadas.

```
rm(list=ls())
library("ggplot2")
library(scales)

#Funcion que simula el lanzamiento de una moneda
lanzamiento <- function(p){
  resultado <- sample(c(0,1), size = 1, prob =c(1-p,p))
  return(resultado)
}

#Realiza lanzamientos hasta obtener un 1, regresa el numero de lanzamientos
experimento <- function(p){
  l_fallidos <-0
  while(lanzamiento(p) != 1) {
    l_fallidos <- l_fallidos + 1
  }
  return (l_fallidos)
}

#Funcion que repite la funcion experimento(p) N veces
#Regresa un vector con el numero de lanzamientos por cada simulacion
simula_lanzamientos <- function(p,N) {
  return(replicate(N, experimento(p)))
}

#Grafica la funcion de densidad de una distribucion
grafica_densidad <- function(dproba) {
  grafica <- ggplot(dproba, aes(x=x, y=fx, xend = x, yend = 0)) +
    geom_segment(size=1, colour = "darkturquoise") +
    geom_point(size = .5, shape = 1, colour = "darkturquoise") +
    scale_x_discrete(breaks = pretty(0: nrow(dproba), n = 10)) +
    labs(title = "Función de masa") +
    theme_bw() +
    theme(plot.title = element_text(hjust = 0.5))
  return(grafica)
}

#Grafica la comparacion de dos distribuciones
grafica_comparacion <- function(dproba1, dproba2) {
  dproba12 <- rbind(dproba1, dproba2)
  grafica <- ggplot(dproba12, aes(x=x, y=fx, xend = x, yend = 0,
                                  group=group, col=group, fill=group, linetype = group)) +
```

```r
    scale_linetype_manual(values=c(2,1)) +
    scale_colour_manual(values=c("firebrick1", "darkturquoise")) +
    geom_segment(size=1) +
    geom_point(size = .5, shape = 1) +
    scale_x_discrete(breaks = pretty(0: nrow(dproba1), n = 10)) +
    labs(title = "Comparación de distribuciones") +
    theme_bw() +
    theme(plot.title = element_text(hjust = 0.5))
  return(grafica)
}

#Simula una VA geometrica, p es el parametro de G(p)
#N es el numero de repeticiones
simular_experimento <- function(p,N){
  print(paste("Simulación de Geom(p) con parámetros p =", p, ", N =",N ))
  simulacion <- replicate(N, experimento(p))
  print(paste("La media es:", mean(simulacion)))
  print(paste("La desviación estándar es:", sd(simulacion)))
  t_contingencias <- table(factor(simulacion, levels = 0:max(simulacion)))
  t_prop <- prop.table(t_contingencias)
  d_simulacion <- as.data.frame(t_prop)
  x <- 0:max(simulacion)
  d_geom <- data.frame(x, dgeom(x,p))
  names(d_simulacion) <- c("x","fx")
  names(d_geom) <- c("x","fx")
  d_simulacion$group <- "Simulacion(p,N)"
  d_geom$group <- "G(p)"
  gdensidad <- grafica_densidad(d_simulacion) +
    labs(subtitle = paste("p =", p, ", N =",N ))
  gcomparacion <- grafica_comparacion(d_simulacion, d_geom) +
    labs(subtitle = paste("p =", p, ", N =",N ))
  list(gdensidad,gcomparacion)
}
```

```r
#Demostracion de la funcion simula_lanzamientos
p <- .3
N <- 30
simula_lanzamientos(p,N)
```

```
##  [1] 0 5 0 1 1 0 2 0 2 1 2 2 0 0 0 3 1 0 0 1 0 4 0 1 5 0 1 0 0 8
```

Ahora se procede a comparar las simulaciones con la distribución geométrica correspondiente.

```
#Se establecen los parametros de cada simulacion
p <- .5
N <- 10^4
simular_experimento(p, N)
```

```
## [1] "Simulación de Geom(p) con parámetros p = 0.5 , N = 10000"
## [1] "La media es: 1.0152"
## [1] "La desviación estándar es: 1.44553724192488"
```

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

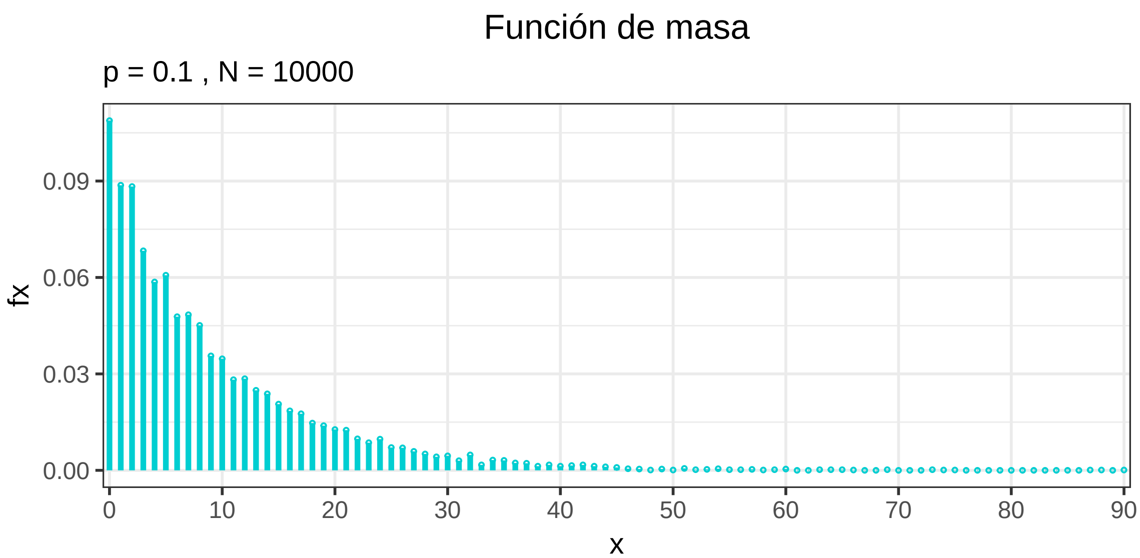

```
##
## [[2]]
```


```
p <- .1
N <- 10^4
simular_experimento(p, N)
```

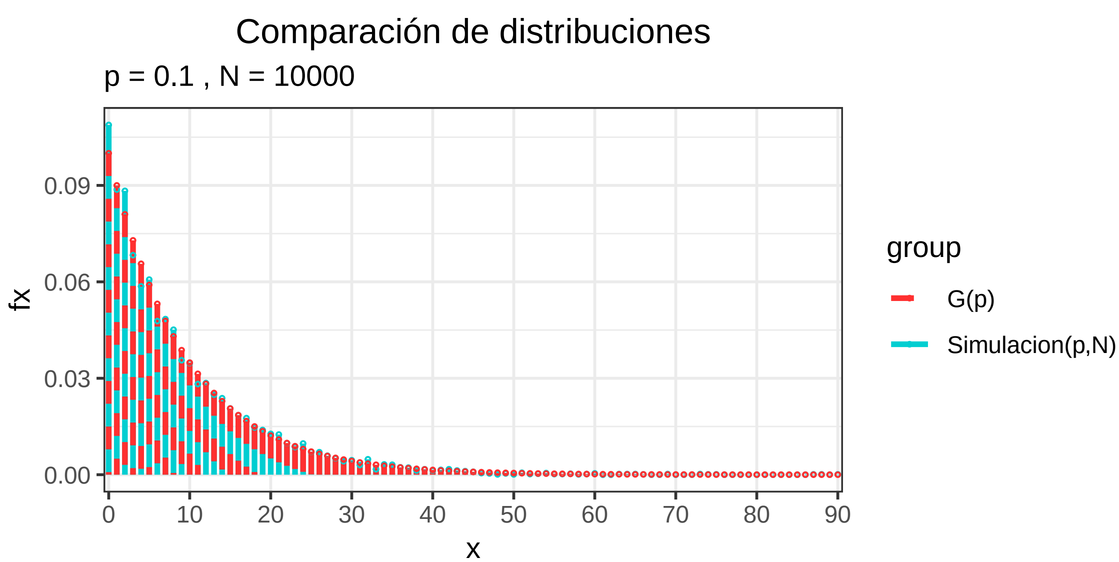
```
## [1] "Simulación de Geom(p) con parámetros p = 0.1 , N = 10000"
```

```
## [1] "La media es: 8.9654"
## [1] "La desviación estándar es: 9.48737075592527"
```

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



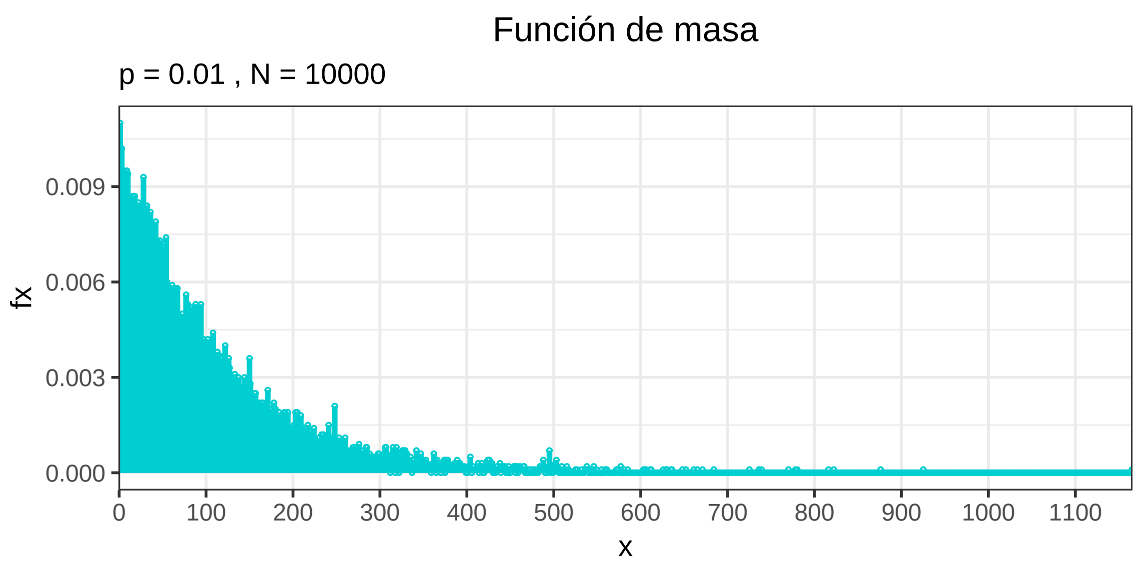

```
##
## [[2]]
```



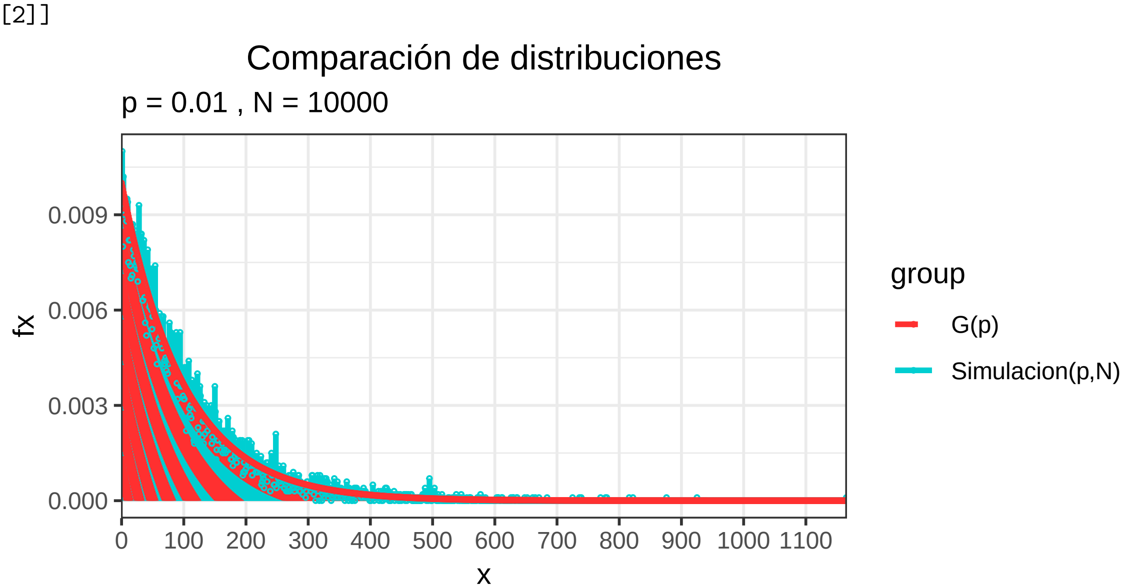

```r
p <- .01
N <- 10^4
simular_experimento(p, N)
```

```
## [1] "Simulación de Geom(p) con parámetros p = 0.01 , N = 10000"
## [1] "La media es: 99.5156"
## [1] "La desviación estándar es: 100.242965947054"
```

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



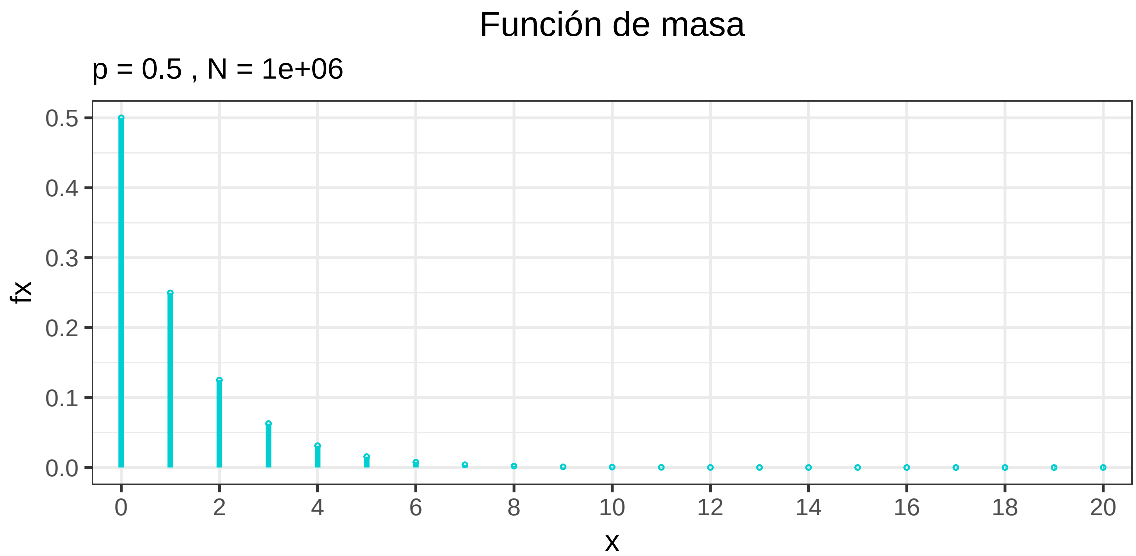

```
##
## [[2]]
```




```r
p <- .5
N <- 10^6
simular_experimento(p, N)
```

```
## [1] "Simulación de Geom(p) con parámetros p = 0.5 , N = 1e+06"
## [1] "La media es: 0.999332"
## [1] "La desviación estándar es: 1.41246506110188"
```

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

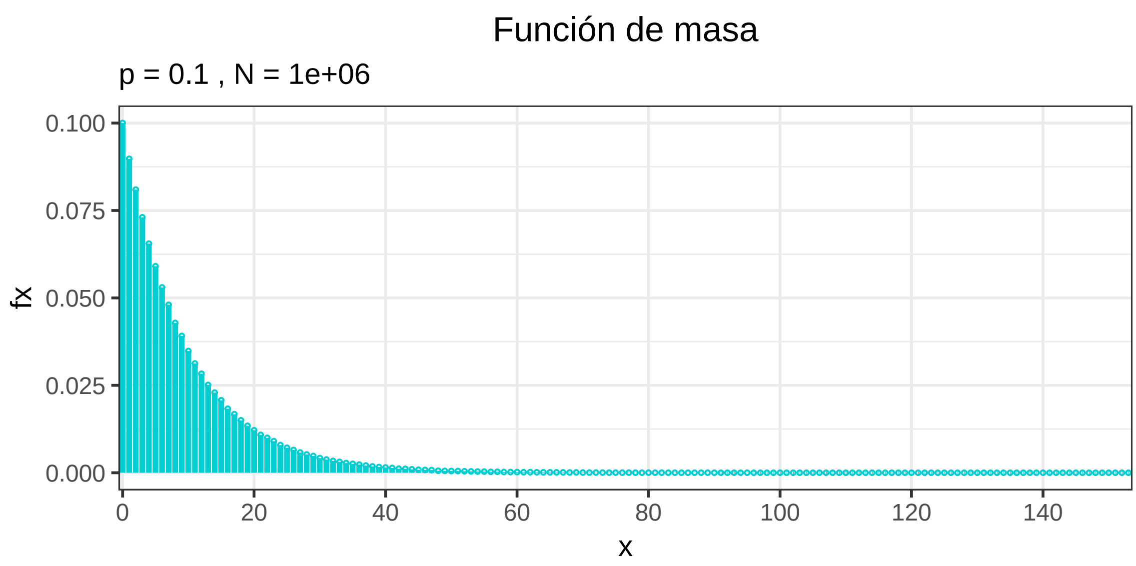



```
##
## [[2]]
```

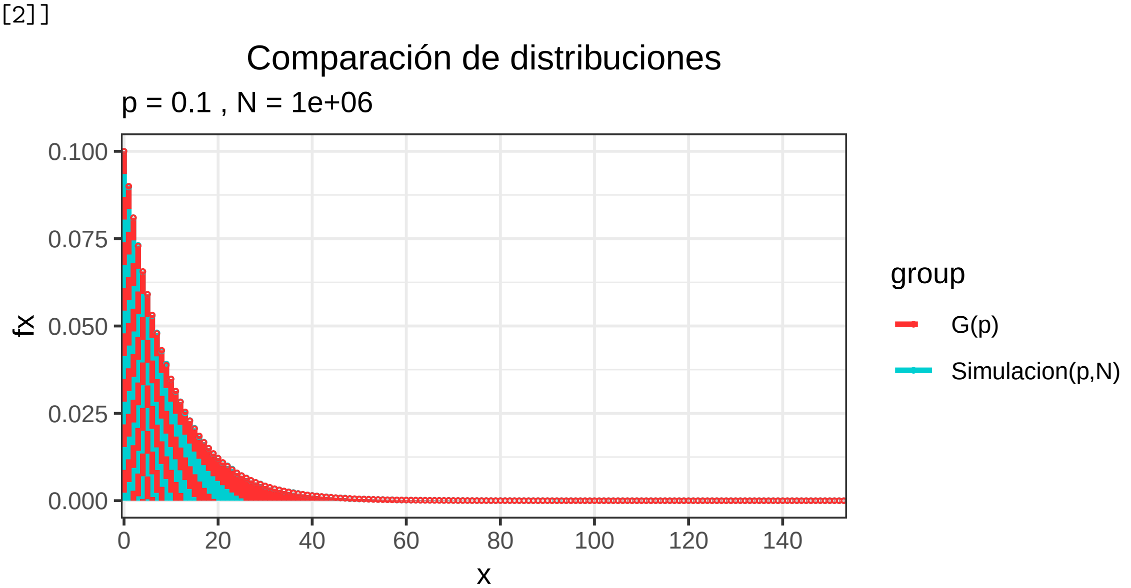

```r
p <- .1
N <- 10^6
simular_experimento(p, N)
```

```
## [1] "Simulación de Geom(p) con parámetros p = 0.1 , N = 1e+06"
## [1] "La media es: 9.008566"
## [1] "La desviación estándar es: 9.50088011135637"
```

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



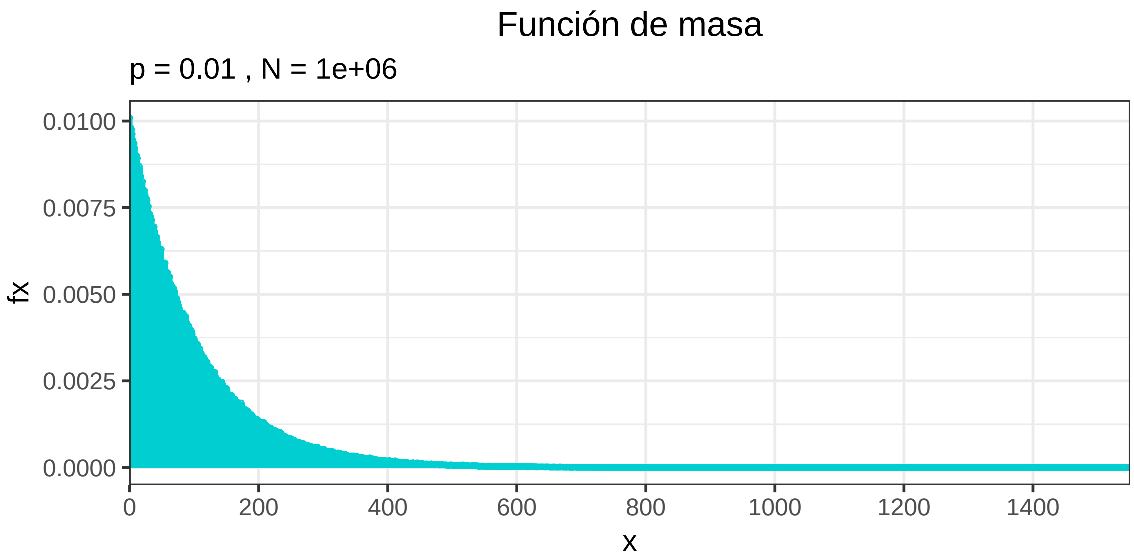

```
##
## [[2]]
```



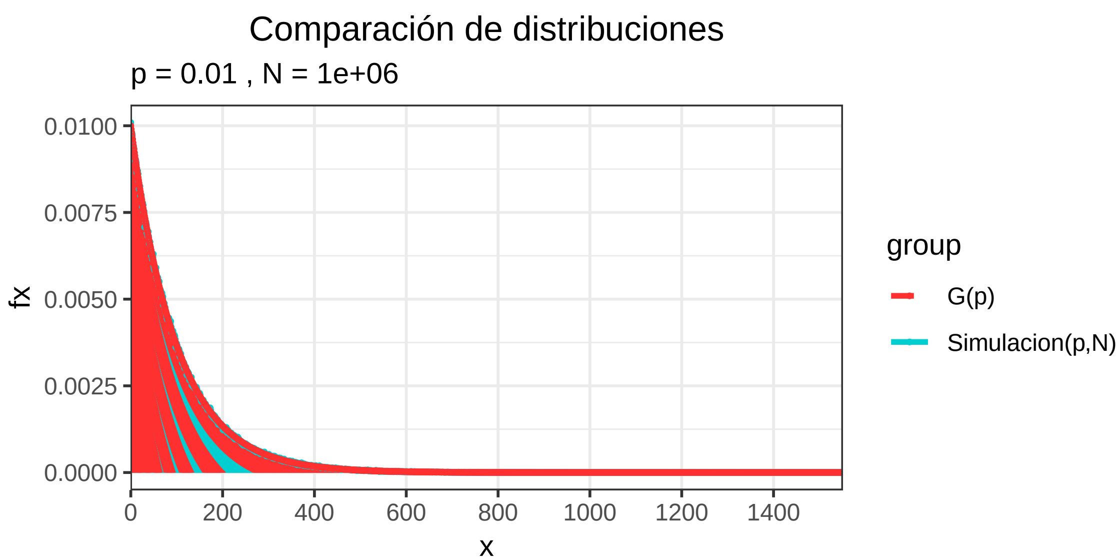

```r
p <- .01
N <- 10^6
simular_experimento(p, N)
```

```
## [1] "Simulación de Geom(p) con parámetros p = 0.01 , N = 1e+06"
## [1] "La media es: 99.089427"
## [1] "La desviación estándar es: 99.6189558655783"
```

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




```
##
## [[2]]
```



Ejercicio 5

Usando las ideas del inciso anterior escriba una función en R que simule N veces los lanzamientos de moneda hasta obtener r águilas. La función deberá recibir como parámetros a la probabilidad p de obtener águila, al número r de águilas a observar antes de detener el experimento y al número N de veces que se repite el experimento; y tendrá que regresar un vector de longitud N que contenga el número de lanzamientos hasta obtener las r águilas en cada uno de los N experimentos. Grafique las frecuencias normalizadas de los experimentos para $N = 10_6$, $p = 0, 2; 0, 1$ y $r = 2; 7$ y compárelos contra la función de masa de la distribución más adecuada para modelar este tipo de experimentos.

A continuación se presenta el código desarrolado en R para este ejercicio, la función simula_lanzamientos(p,r,N) regresa un vector con las características deseadas.

```
rm(list=ls())
library("ggplot2")
library(scales)
```

```r
#Funcion que simula el lanzamiento de una moneda
lanzamiento <- function(p){
  resultado <- sample(c(0,1), size = 1, prob =c(1-p,p))
  return(resultado)
}
#Realiza lanzamientos hasta obtener r 1s
#Regresa el número de lanzamientos realizados
experimento <- function(p,r){
  l_totales <- 0
  l_acertados <-0
  while(l_acertados < r ) {
    if(lanzamiento(p) == 1) {
      l_acertados <- l_acertados + 1
    }
    l_totales <- l_totales + 1
  }
  return (l_totales)
}

#Funcion que repite la funcion experimento(p) N veces
#Regresa un vector con el numero de lanzamientos por cada simulacion
simula_lanzamientos <- function(p,r,N) {
  return(replicate(N, experimento(p,r)))
}

#Grafica la funcion de densidad de una distribucion
grafica_densidad <- function(dproba) {
  grafica <- ggplot(dproba, aes(x=x, y=fx, xend = x, yend = 0)) +
    geom_segment(size=1, colour = "darkturquoise") +
    geom_point(size = .5, shape = 1, colour = "darkturquoise") +
    scale_x_discrete(breaks = pretty(0: nrow(dproba), n = 10)) +
    labs(title = "Función de masa") +
    theme_bw() +
    theme(plot.title = element_text(hjust = 0.5))
  return(grafica)
}

#Grafica la comparacion de dos distribuciones
grafica_comparacion <- function(dproba1, dproba2) {
  dproba12 <- rbind(dproba1, dproba2)
  grafica <- ggplot(dproba12, aes(x=x, y=fx, xend = x, yend = 0,
                                  group=group, col=group, fill=group, linetype = group)) +
    scale_linetype_manual(values=c(2,1)) +
    scale_colour_manual(values=c("firebrick1", "darkturquoise")) +
    geom_segment(size=1) +
    geom_point(size = .5, shape = 1) +
    scale_x_discrete(breaks = pretty(0: nrow(dproba1), n = 10)) +
    labs(title = "Comparación de distribuciones") +
    theme_bw() +
    theme(plot.title = element_text(hjust = 0.5))
  return(grafica)
}
```

```r
#Simula una VA binomial negativa BN(p,r)
#N es el numero de repeticiones
simular_experimento <- function(p,r,N){
  print(paste("Simulación de BN(p,r) con parámetros p =", p,", r =", r,", N =",N ))
  simulacion <- replicate(N, experimento(p,r))
  print(paste("La media es:", mean(simulacion)))
  print(paste("La desviación estándar es:", sd(simulacion)))
  t_contingencias <- table(factor(simulacion, levels = 0:max(simulacion)))
  t_prop <- prop.table(t_contingencias)
  d_simulacion <- as.data.frame(t_prop)
  x <- 0:max(simulacion)
  d_geom <- data.frame(x, dnbinom(x-r,size= r, prob = p)) #reparametrizacion
  names(d_simulacion) <- c("x","fx")
  names(d_geom) <- c("x","fx")
  d_simulacion$group <- "Simulacion(p,r,N)"
  d_geom$group <- "BN(p,r)"
  gdensidad <- grafica_densidad(d_simulacion) +
    labs(subtitle = paste("p =", p,", r =", r,", N =",N ))
  gcomparacion <- grafica_comparacion(d_simulacion, d_geom) +
    labs(subtitle = paste("p =", p,", r =", r,", N =",N ))
  list(gdensidad,gcomparacion)
}
```

```r
#Demostracion de la funcion simula_lanzamientos
p <- .2
r <- 2
N <- 30
simula_lanzamientos(p,r,N)
```

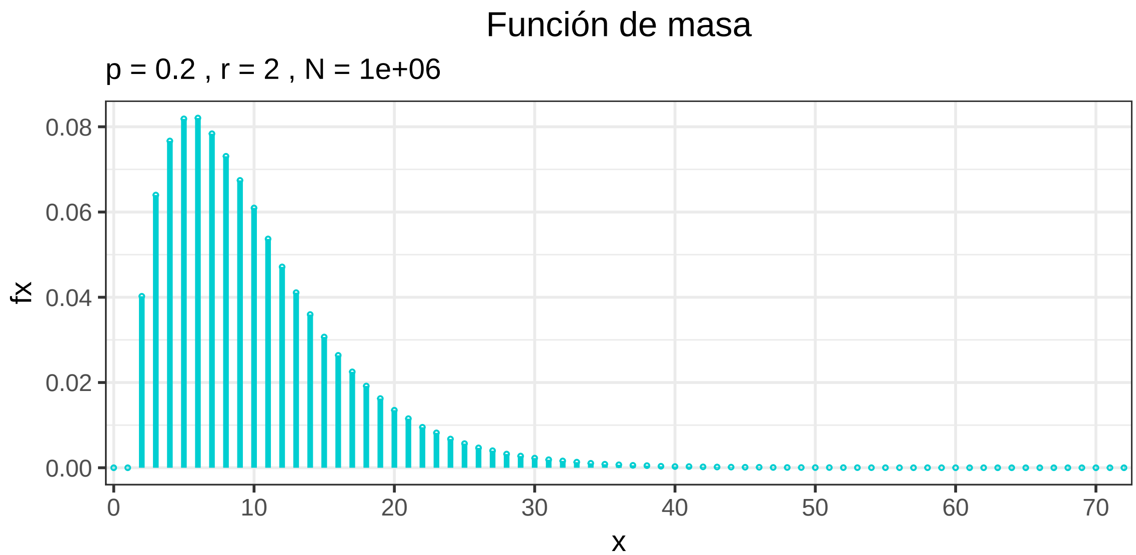
```
##  [1]  6  4  5 24  9 12 12  8  3  5  6  7  4  4  6  6 14 18 11 11  6  5  8
## [24] 17  9 19  6  9 15  5
```

Ahora se procede a comparar las simulaciones con la distribución binomial negativa correspondiente.

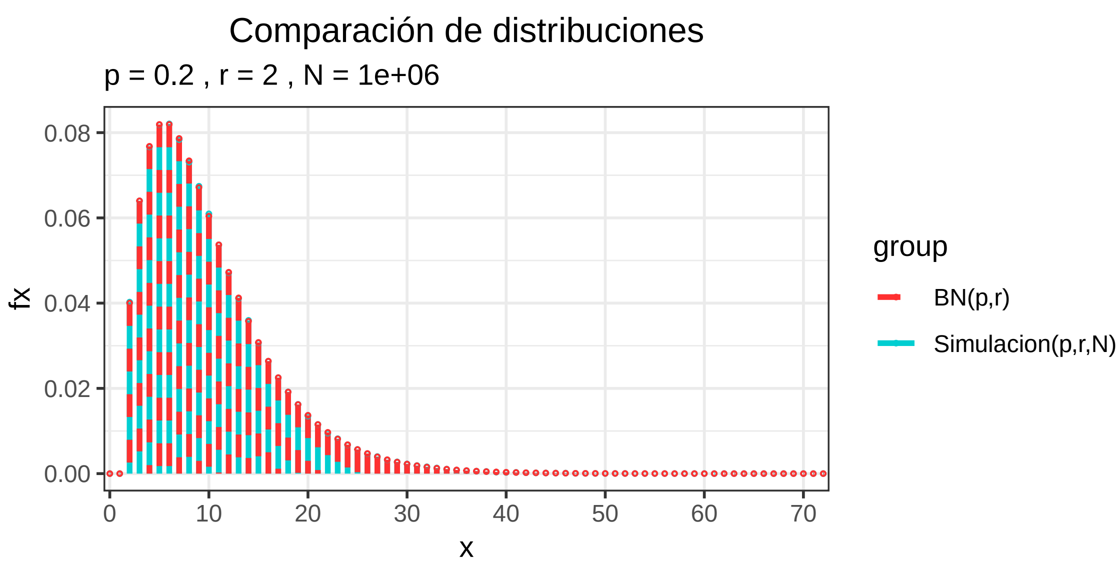
```
#Se establecen los parametros de cada simulacion
p <- .2
r <- 2
N <- 10^6
simular_experimento(p,r,N)
```

```
## [1] "Simulación de BN(p,r) con parámetros p = 0.2 , r = 2 , N = 1e+06"
## [1] "La media es: 9.99636"
## [1] "La desviación estándar es: 6.32066220432406"
```

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

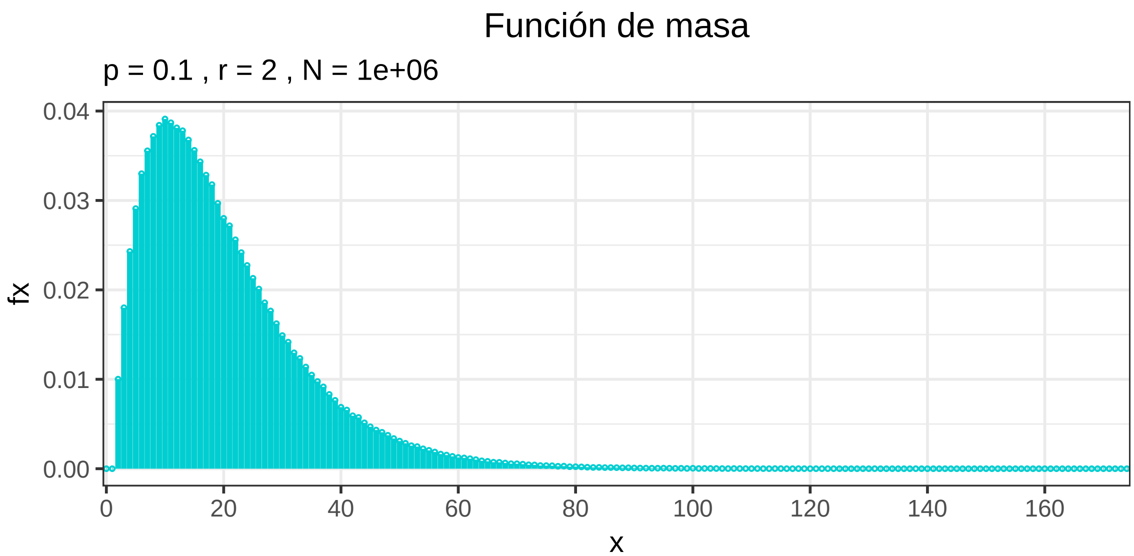



```
##
## [[2]]
```

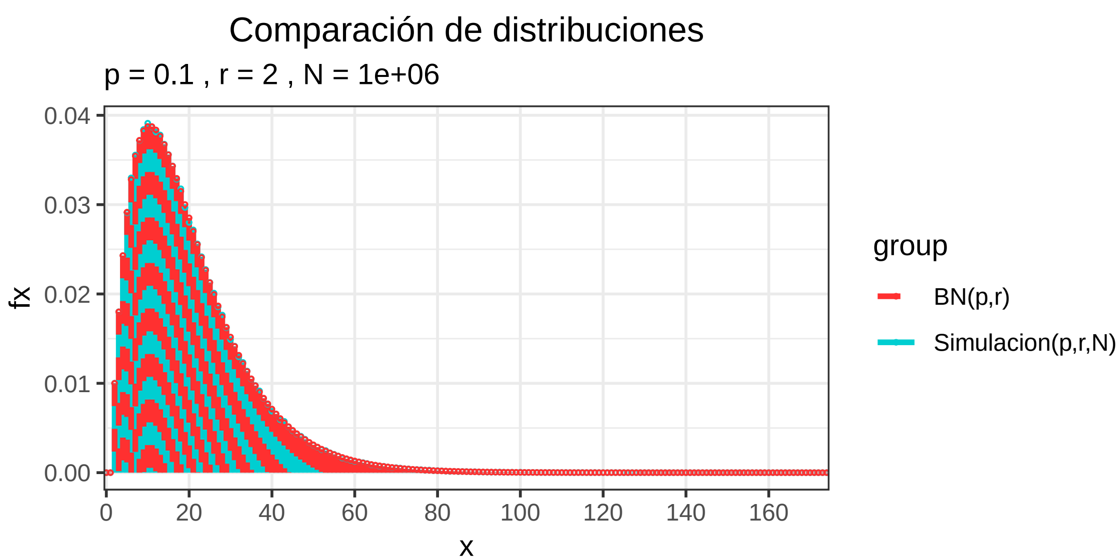



```
p <- .1
r <- 2
N <- 10^6
simular_experimento(p,r,N)
```

```
## [1] "Simulación de BN(p,r) con parámetros p = 0.1 , r = 2 , N = 1e+06"
## [1] "La media es: 19.98478"
## [1] "La desviación estándar es: 13.4164390339775"

## [[1]]
```

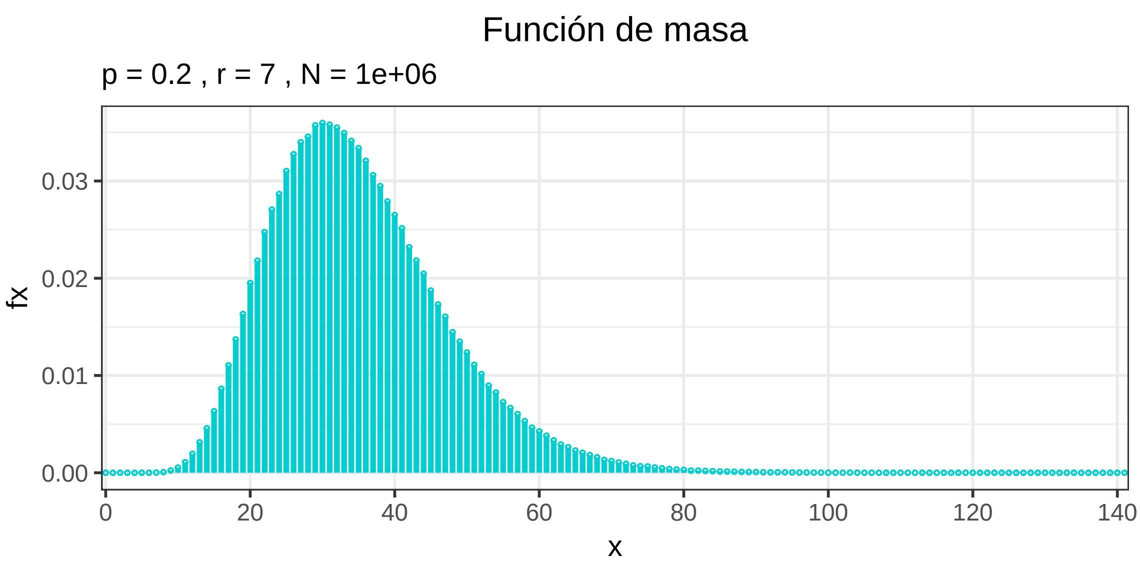



```
##
## [[2]]
```

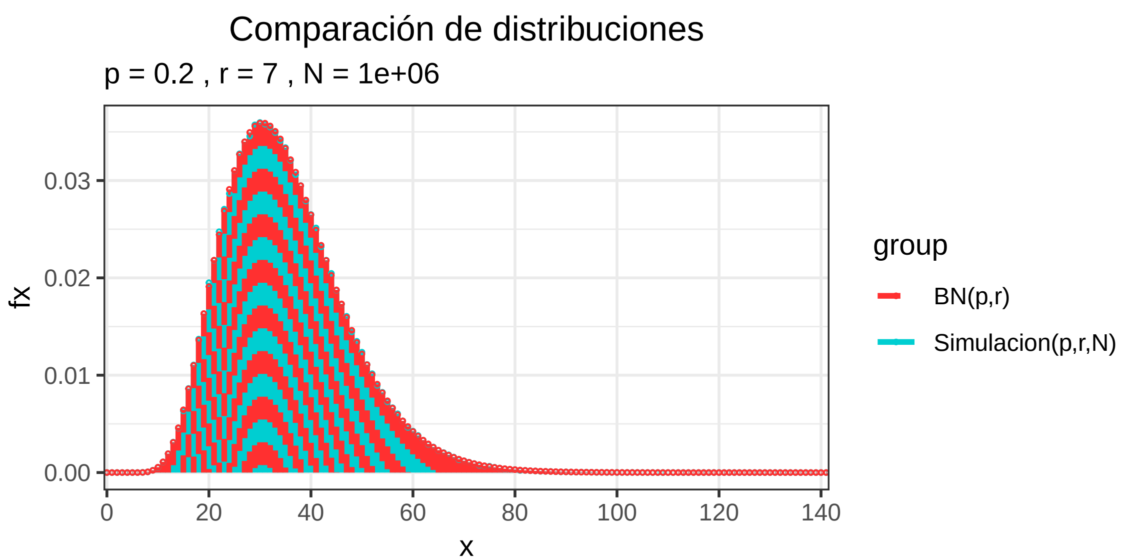



```
p <- .2
r <- 7
N <- 10^6
simular_experimento(p,r,N)
```

```
## [1] "Simulación de BN(p,r) con parámetros p = 0.2 , r = 7 , N = 1e+06"
## [1] "La media es: 34.995365"
## [1] "La desviación estándar es: 11.8284854241717"

## [[1]]
```



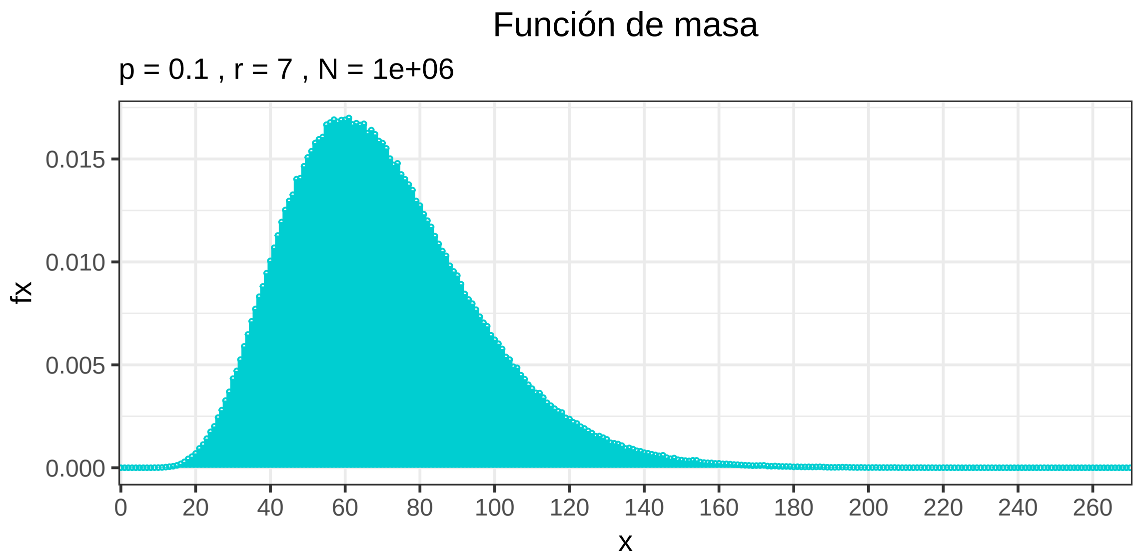

```
##
## [[2]]
```



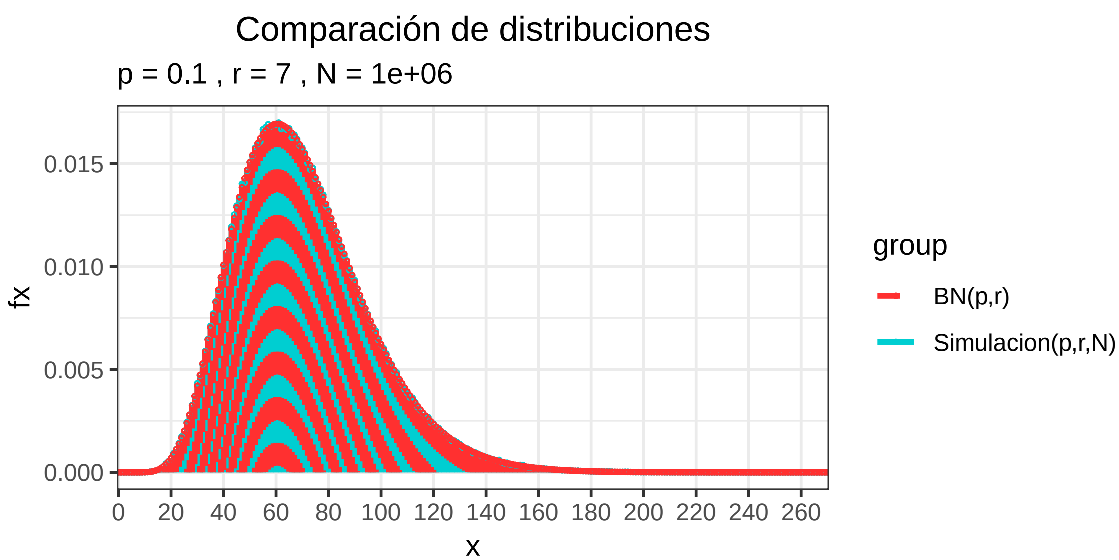

```r
p <- .1
r <- 7
N <- 10^6
simular_experimento(p,r,N)
```

```
## [1] "Simulación de BN(p,r) con parámetros p = 0.1 , r = 7 , N = 1e+06"
## [1] "La media es: 69.973299"
## [1] "La desviación estándar es: 25.1400554509927"
```

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




```
##
## [[2]]
```



Ejercicio 6

Considera X una v.a. con función de distribución F y función de densidad f, y sea A un intervalo de la línea real $\mathbb{R}$. Definimos la función indicadora $1_A(x)$:

$$1_A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{en otro caso} \end{cases}$$

Sea $Y = 1_A(X)$. Encuentre una expresión para la distribución acumulada de Y.

Para calcular la función de distribución acumulada de Y se procede a encontrar primero su función de masa. Note que Y es una variable discreta que cuyo soporte está contenido en $\{0, 1\}$, por lo que que basta calcular $P_Y(0)$ y $P_Y(1)$ para determinar la distribución de Y.

En primer lugar suponga que la cerradura del intervalo A es [a,b]

$$
\begin{aligned}
P_Y(0) &= P(Y=0) && (11)\\
&= P(1_A X = 0) && (12)\\
&= P(X \notin A) && (13)\\
&= 1 - P(X \in A) && (14)\\
&= 1 - P(a \leq X \leq b) && (15)\\
&= 1 - \int_a^b f_X(x)dx && (16)\\
&= 1 - (F_X(b) - F_X(a)) && (17)
\end{aligned}
$$

Ahora para $Y = 1$, se tiene que

$$
\begin{aligned}
P_Y(1) &= P(Y=1) && (18)\\
&= P(1_A X = 1) && (19)\\
&= P(X \in A) && (20)\\
&= P(a \leq X \leq b) && (21)\\
&= \int_a^b f_X(x)dx && (22)\\
&= (F_X(b) - F_X(a)) && (23)
\end{aligned}
$$

De modo que f_Y es tal que

$$
f_Y(y) = \begin{cases} 1 - (F_X(b) - F_X(a)) & \text{si } y = 0 \\ F_X(b) - F_X(a) & \text{si } y = 1 \\ 0 & \text{en otro caso} \end{cases}
$$

Esto a su vez, demuestra que $\sum_{y:f_Y(y)>0} f_Y(y) = 1$.

Ahora se procede a calcular F_y. Si $y < 0$, entonces

$$
F_Y(y) = \sum_{k \leq y: f_Y(k)>0} f_Y(k),
$$

pero para $k < 0$, $f_Y(k) = 0$, por eso, $F_Y(y) = 0$ si $y < 0$.

Si $y = 0$, entonces

$$
F_Y(0) = 1 - (F_X(b) - F_X(a)),
$$

pues ya se vio que si $k < 0$, entonces $f_Y(k) = 0$, por lo cual $\sum_{k \leq 0: f_Y(k)>0} f_Y(k) = f_Y(0) = 1-(F_X(b)-F_X(a))$.

Si $y = 1$, entonces

$$
F_Y(1) = P(Y \leq 1) = P(Y \leq 0) + P(Y = 1),
$$

que a partir de los resultados anteriores, se tiene que $F_Y(1) = 1 - (F_X(b) - F_X(a)) + (F_X(b) - F_X(a)) = 1$.

Finalmente, si $y > 1$, entonces

$$
F_Y(1) \leq F_Y(y) \leq 1,
$$

y como $F_Y(1) = 1$, entonces $F_Y(y) = 1$.

Así pues, F_Y queda descrita como

$$
F_Y(y) = \begin{cases} 0 & \text{si } y < 0 \\ 1 - (F_X(b) - F_X(a)) & \text{si } y = 0 \\ 1 & \text{si } 1 < y \end{cases}
$$

Ejercicio 7

Entre las más famosas de todas las lluvias de meteoros están las Perseidas, que ocurren cada año a principios de agosto. En algunas áreas, la frecuencia de Perseidas visibles promedian seis por cada cuarto de hora. El modelo de probabilidad que describe a Y, el número de meteoros que una persona ve en un cuarto de hora, tiene la función de probabilidad

$$f_Y(y) = \frac{e^{-6}6^y}{y!} \qquad\qquad y = 0, 1, ...$$

Encuentre la probabilidad de que una persona vea en un cuarto de hora determinado al menos la mitad de los meteoros que esperaría ver.

De acuerdo a la espeficicación del problema, se tiene que $Y \sim Poisson(6)$. Con dicha consideración, el número de meteoros que esperaría ver es $E(Y)$, que para este caso es el parámetro $\lambda = 6$, la mitad de esta cantidad es 3, por lo que el problema se resuelve al calcular $P(Y \geq 3)$.

$$\begin{aligned} P(Y \geq 3) &= 1 - P(Y < 3) && (24)\\ &= 1 - \big(P(Y = 0) + P(Y = 1) + P(Y = 2)\big) && (25)\\ &= 1 - (0.002478752 + 0.01487251 + 0.04461754) && (26)\\ &= 0.9380312 && (27) \end{aligned}$$

Así pues

$$\mathbf{P(Y \geq 3) = 0.9380312}$$

Ejercicio 8

Las calificaciones de un estudiante de primer semestre en un examen de química se describen por la densidad de probabilidad

$$f_Y(y) = 6y(1 - y) \qquad\qquad 0 \leq y \leq 1$$

donde y representa la proporción de preguntas que el estudiante contesta correctamente. Cualquier calificación menor a 0.4 es reprobatoria. Responda lo siguiente:

¿Cuál es la probabilidad de que un estudiante repruebe?

Si 6 estudiantes toman el examen, ¿cuál es la probabilidad de exactamente 2 reprueben?

Nota: Para este ejercicio agilizará algunos cálculos el saber que

$$\int 6y(1 - y)dy = 3y^2 - 2y^3 + C$$

En primer lugar, se observa que $\int_{-\infty}^{\infty} f_Y(y)dy = 3(1)^2 - 2(1)^3 = 1$.

La probabilidad de reprobar el examen está dada por $P(Y \leq .4)$ que es igual a $3(.4)^2 - 2(.4)^3 = 0.32 - 0.128 = 0.352$.

Así, la probabilidad de reprobar es **0.352**

La probabilidad de que al presentar 6 estudiantes el examen, dos de ellos reprueben, está dada por una variable $X \sim B(6, 0.352)$, pues el resultado de cada examen es independiente del resto.

Se desea conocer $P(X = 6)$, que es igual a

$$P(X = 2) = \binom{6}{2}(.352)^2(.648)^4 = \mathbf{0.3277001}$$

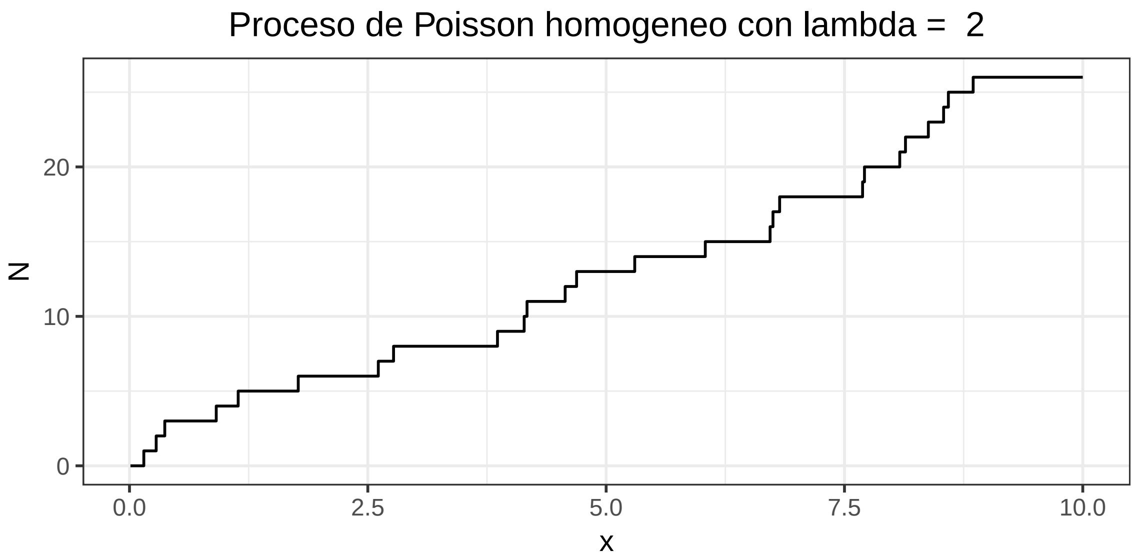
Que es el resultado requerido.

Ejercicio 9

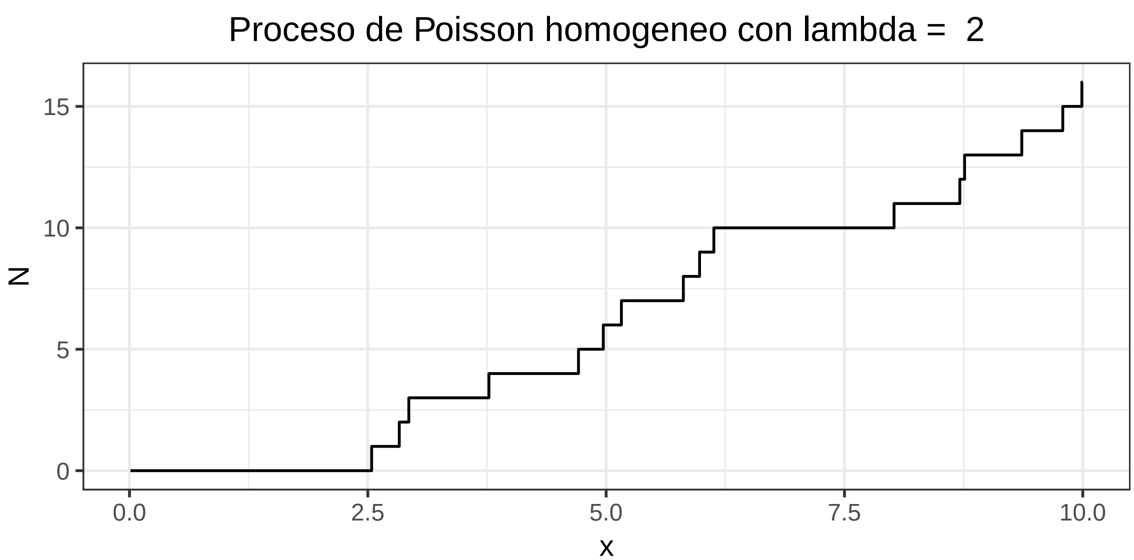
Escriba una función en R que simule una aproximación al proceso Poisson a partir de las 5 hipótesis que usamos en clase para construir tal proceso. Usando esta función, simule tres trayectorias de un proceso Poisson $\lambda = 2$ sobre el intervalo $[0, 10]$ y grafíquelas. Además, simule 10^4 veces un proceso de Poisson N con $\lambda = \frac{1}{2}$ y hasta el tiempo $t = 1$. Haga un histograma de $N(1)$ en su simulación anterior y compare contra la distribución de Poisson correspondiente.\ Hint: Considere el intervalo $[0, T]$ y un número real positivo dt que sea mucho más pequeño que la longitud de $[0, T]$ y que divida dicha longitud, digamos $T = dt = 1000$. Divida el intervalo $[0, T]$ en intervalitos de longitud dt que tengan la forma $(k * dt, (k + 1) * dt]$, $k = 0, 1, 2, \ldots, (T = dt * 1)$. Para cada uno de estos intervalitos simule una v.a. $Bernoulli(\lambda * dt + 10^6)$ y guarde su resultado en un vector del tamaño adecuado.

```r
rm(list=ls())
library("ggplot2")

#Funcion que simula el lanzamiento de una moneda
lanzamiento <- function(p){
  resultado <- sample(c(0,1), size = 1, prob =c(1-p,p))
  return(resultado)
}
#Simula un proceso de Poisson a través de subdivisiones
simulacion_poisson <- function(lambda, x_max, n_divisiones) {
  longitud <- x_max / n_divisiones
  p <- lambda * longitud + 10^-6
  simulacion <- replicate(n_divisiones, lanzamiento(p))
  s_acumulada <- data.frame((1:n_divisiones)*longitud, cumsum(simulacion))
  names(s_acumulada) <- c("x", "N")
  return(s_acumulada)
}

#Muestra la trayectoria de un proceso de Poisson
trayectoria_poisson <- function(lambda, x_max, n_divisiones) {
  s_acumulada <- simulacion_poisson(lambda, x_max, n_divisiones)
  ggplot() +
    geom_step(data=s_acumulada, mapping=aes(x=x, y=N)) +
    labs(title = paste("Proceso de Poisson homogeneo con lambda = ", lambda)) +
    theme_bw() +
    theme(plot.title = element_text(hjust = 0.5))
}
```

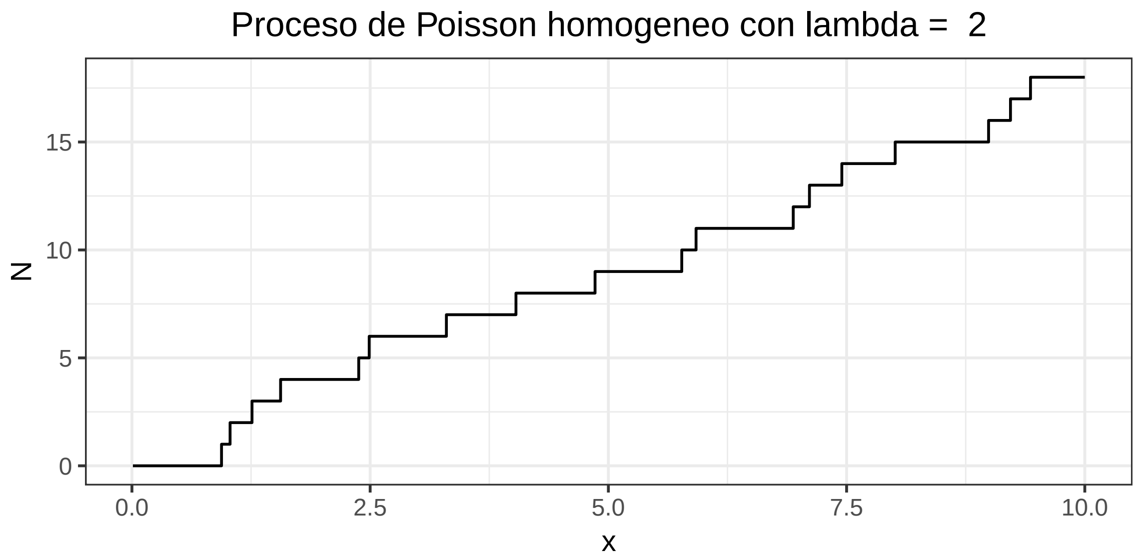
```r
#Parámetros
lambda <- 2
x_max <- 10
n_divisiones <- 1000
#Trayectoria1
trayectoria_poisson(lambda,x_max, n_divisiones)
```




```
#Trayectoria2
trayectoria_poisson(lambda,x_max, n_divisiones)
```




```
#Trayectoria3
trayectoria_poisson(lambda,x_max, n_divisiones)
```


Proceso de Poisson homogeneo con lambda = 2
N
0
5
10
15
x
0.0
2.5
5.0
7.5
10.0