

PRÁCTICA 3: DISTRIBUCIONES DE PROBABILIDAD EN R

Alba Calvo Herrero

1ºB GII

22 de noviembre de
2019

EJERCICIO 1. Representar la función Binomial para los siguientes casos B1(0.3,30); B2(0.6,30) y B3(0.8,100).

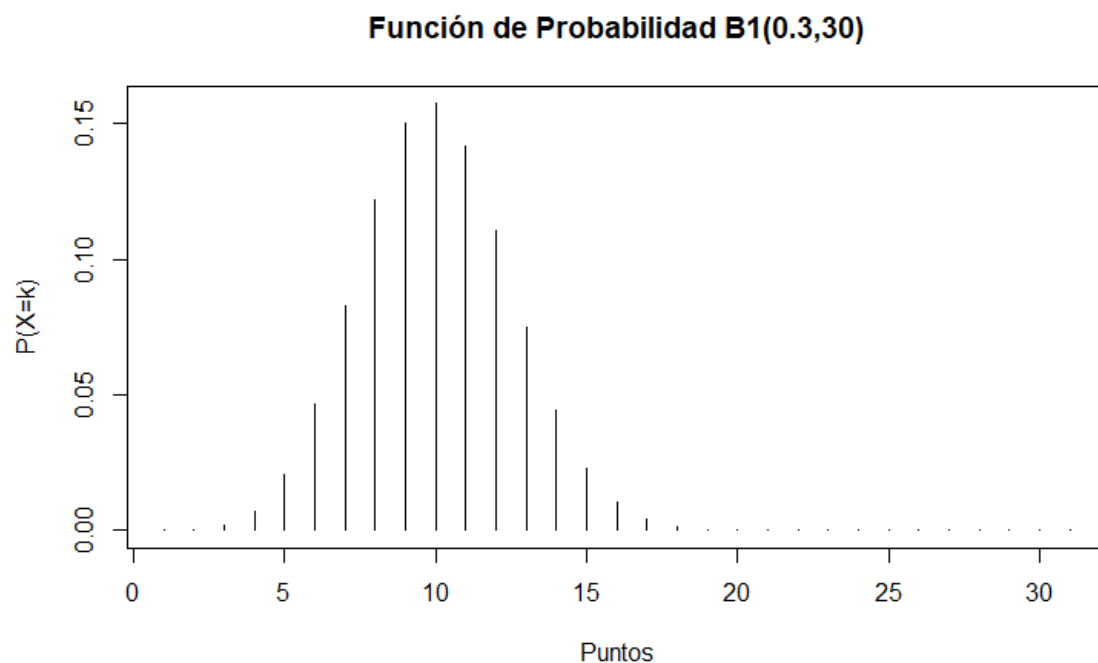
Para empezar, sabemos que es una distribución binomial por lo que usaremos el comando ***dbinom*** (***d*** porque queremos una función de probabilidad y ***binom*** por la distribución binomial).

Para representar una función de distribución de probabilidad lo primero que tenemos que tener es un conjunto de datos para la variable de entrada de la función. Por eso, en el primer caso tomaremos valores de 0 a 30 (0:30), ya que, sabemos que $n=30$.

Para representar funciones binomiales se utiliza el comando ***plot*** de la siguiente manera:

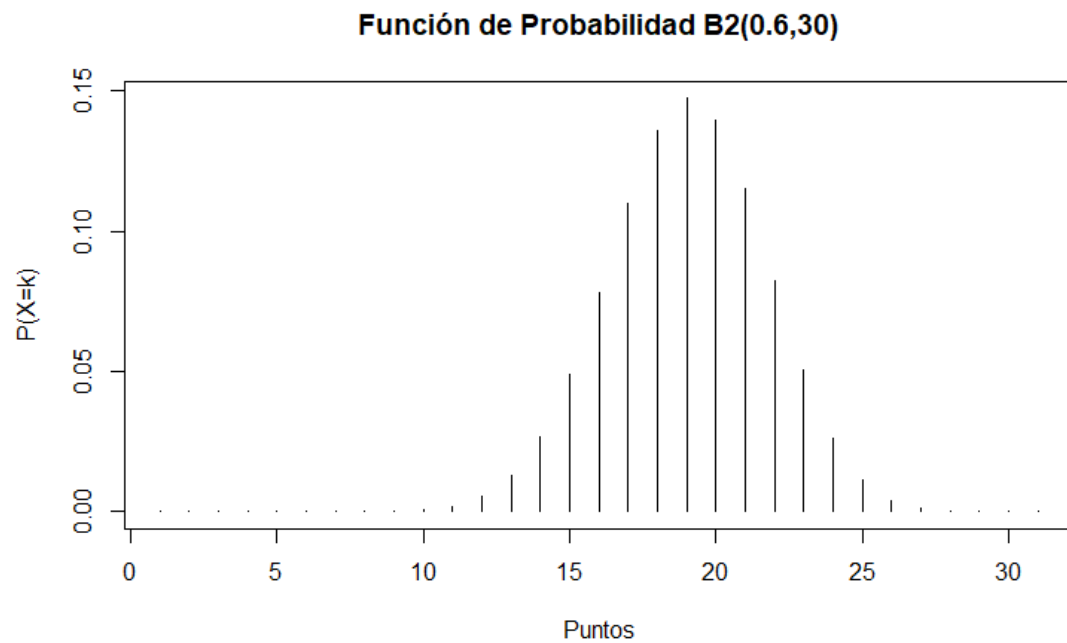
Para B1(0.3,30):

```
> plot(dbinom(0:30,30,0.3),type="h",xlab="Puntos",ylab="P(X=k)",main="Función de Probabilidad B1(0.3,30)")
```



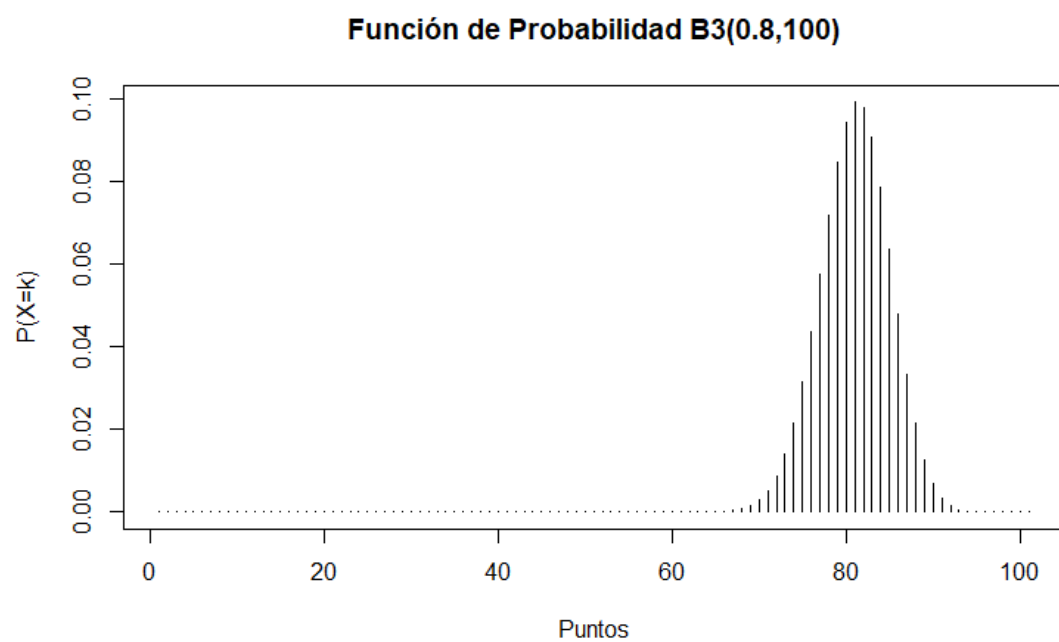
Para B2(0.6,30):

```
> plot(dbinom(0:30,30,0.6),type="h",xlab="Puntos",ylab="P(X=k)",main="Función de Probabilidad B2(0.6,30)")
```



Y para B3(0.8,100):

```
> plot(dbinom(0:100,100,0.8),type="h",xlab="Puntos",ylab="P(X=k)",main="Función de Probabilidad B3(0.8,100)")
```



EJERCICIO 2. Para cada una de las funciones de distribución de probabilidad:

- a. Calcular la probabilidad de que la variable tome el valor 20.

Si x sigue una distribución binomial $B(n,p)$, entonces $P(X=k)=dbinom(k,n,p)$.

Por lo tanto, la probabilidad será:

- En B1: $dbinom(20,30,0.3)$
- En B2: $dbinom(20,30,0.6)$
- En B3: $dbinom(20,100,0.8)$

```
> #a. P(x=20)
> dbinom(20,30,0.3)
[1] 2.959225e-05
> dbinom(20,30,0.6)
[1] 0.1151854
> dbinom(20,100,0.8)
[1] 7.470518e-38
```

- b. Calcular la probabilidad de que la variable tome un valor igual o menor de 10.

Sabiendo que $P(X \leq k)=pbinom(k,n,p)$, la probabilidad de x menor o igual que 10 será:

- En B1: $pbinom(10,30,0.3)$
- En B2: $pbinom(10,30,0.6)$
- En B3: $pbinom(10,100,0.8)$

```
> #b. P(x<=10)
> pbinom(10,30,0.3)
[1] 0.7303704
> pbinom(10,30,0.6)
[1] 0.002853883
> pbinom(10,100,0.8)
[1] 2.365727e-51
```

- c. Calcular la probabilidad de que la variable tome un valor mayor de 25.

En este caso nos pide la probabilidad de x mayor que un número k . Debemos saber que es lo mismo que el complementario de x menor que el número, es decir, $P(X > k) = 1 - P(X \leq k) = 1 - pbinom(k,n,p)$.

Por lo tanto, estos serían los comandos para cada distribución:

- En B1: $1-pbinom(25,30,0.3)$
- En B2: $1-pbinom(25,30,0.6)$
- En B3: $1-pbinom(25,100,0.8)$

```

> #c.  $P(x > 25) = 1 - P(x < 25)$ 
> 1-pbinom(25,30,0.3)
[1] 1.783748e-10
> 1-pbinom(25,30,0.6)
[1] 0.001510074
> 1-pbinom(25,100,0.8)
[1] 1

```

EJERCICIO 3. Representar la función normal, con 50 números generados aleatoriamente, con los siguientes datos de medias y desviaciones típicas: (0,1); (0,0.5) y (0,2). Para cada una de estas funciones calcular:

- a. El valor de la función de probabilidad en el punto 0.2.
En este ejercicio tenemos que trabajar con distribuciones normales. Por eso, usaremos el comando **dnorm** (**d** por la función de probabilidad y **norm** por la distribución normal).

Queremos saber el valor que toma la función de probabilidad para el punto 0.2. Para ello, escribimos: *dnorm(punto,media,desviación típica)*:

- *dnorm(0.2,0,1)*
- *dnorm(0.2,0,0.5)*
- *dnorm(0.2,0,2)*

```

> dnorm(0.2,0,1)
[1] 0.3910427
> dnorm(0.2,0,0.5)
[1] 0.7365403
> dnorm(0.2,0,2)
[1] 0.1984763

```

- b. La probabilidad de que la variable sea mayor que 1.
Al igual que en el anterior ejercicio, la probabilidad de z mayor que un número será el complementario de la probabilidad de z menor que el número, es decir, $P(z > k) = 1 - p(z < k)$.

Por tanto, escribimos:

```

> 1-pnorm(1,0,1)
[1] 0.1586553
> 1-pnorm(1,0,0.5)
[1] 0.02275013
> 1-pnorm(1,0,2)
[1] 0.3085375

```

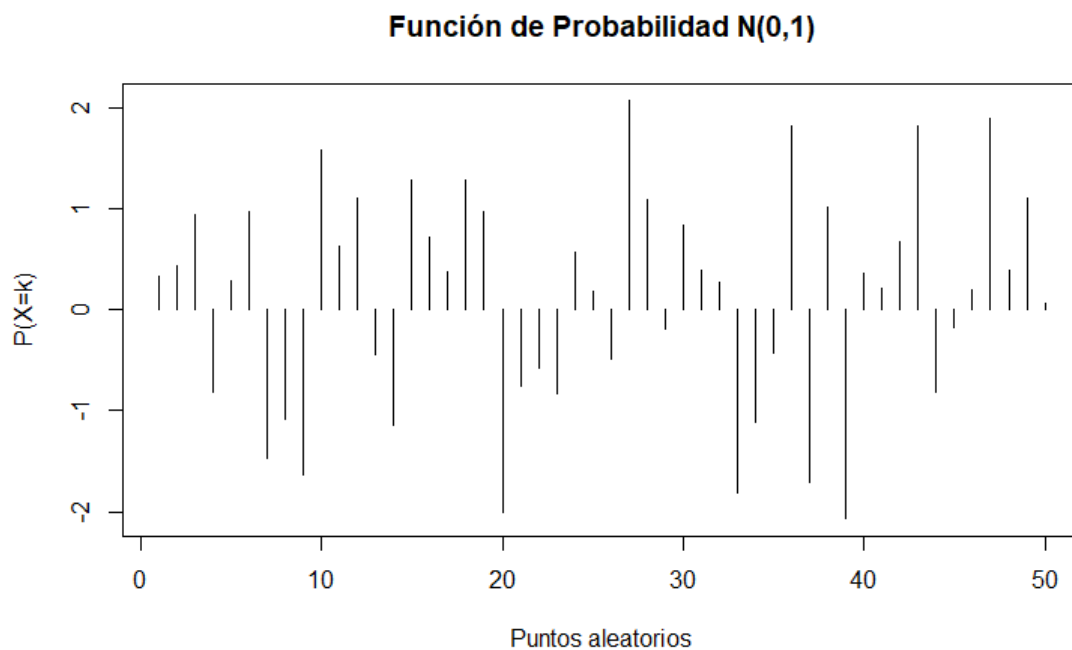
EJERCICIO 4. Representar el histograma de cada una de las secuencias aleatorias del ejercicio 3.

Primero, represento las funciones de probabilidad para cada uno de los casos.

Para ello usamos el comando **plot**. Como nos pide representarlas con 50 numeros generados aleatoriamente usamos el comando **rnorm** (donde **r** es una función para simular datos con dicha distribución normal).

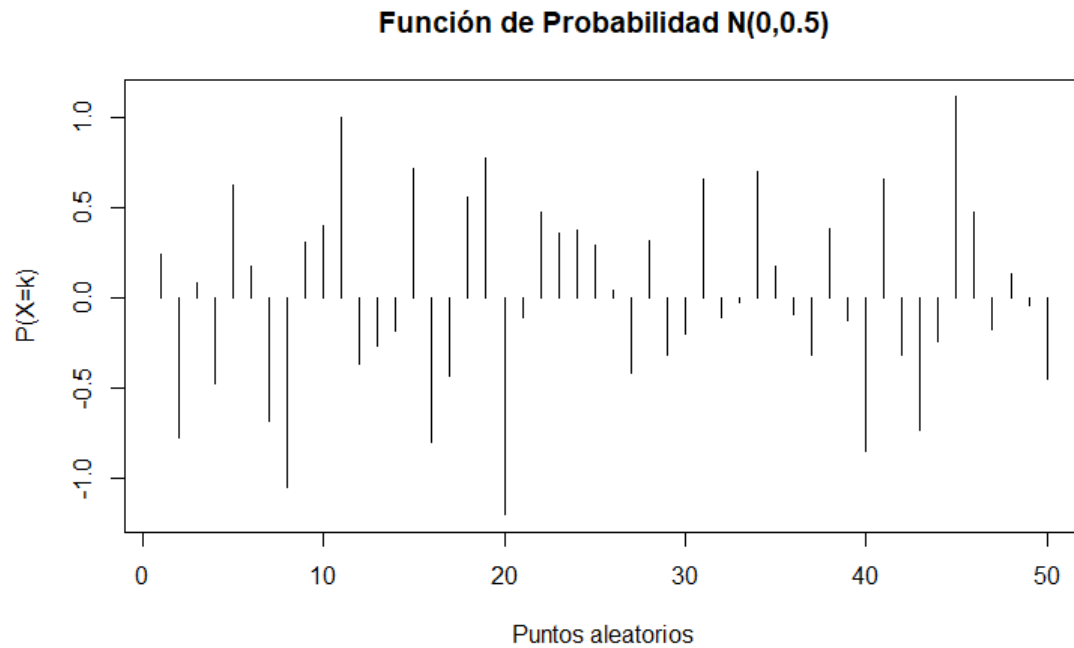
Para la distribución N(0,1):

```
> plot(1:50, rnorm(50,0,1),type="h",xlab="Puntos aleatorios",ylab="P(X=k)",main="Función de Probabilidad N(0,1)")
```



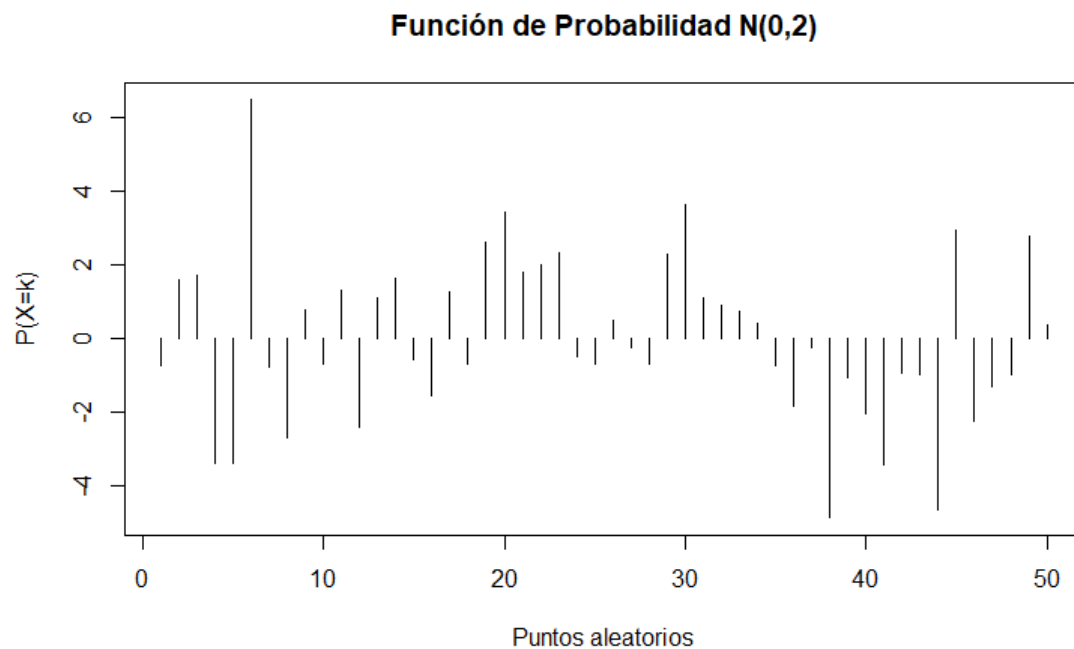
Para la distribución $N(0,0.5)$:

```
> plot(1:50, rnorm(50,0,0.5),type="h",xlab="Puntos aleatorios",ylab="P(X=k)",main="Función de Probabilidad N(0,0.5)")
```



Para la distribución $N(0,2)$:

```
> plot(1:50, rnorm(50,0,2),type="h",xlab="Puntos aleatorios",ylab="P(X=k)",main="Función de Probabilidad N(0,2)")
```

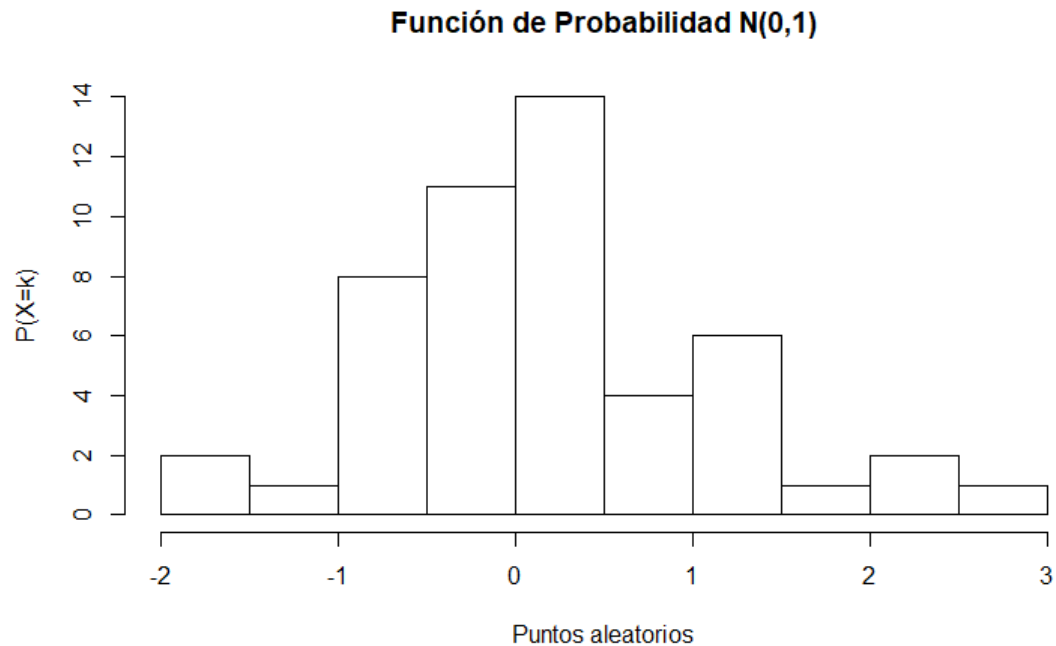


El siguiente paso es representar los histogramas para cada distribución.

Para ello, usamos la función **hist**. Como también queremos generar 50 puntos aleatorios dentro de hist usamos de nuevo la función **rnorm** de la siguiente manera:

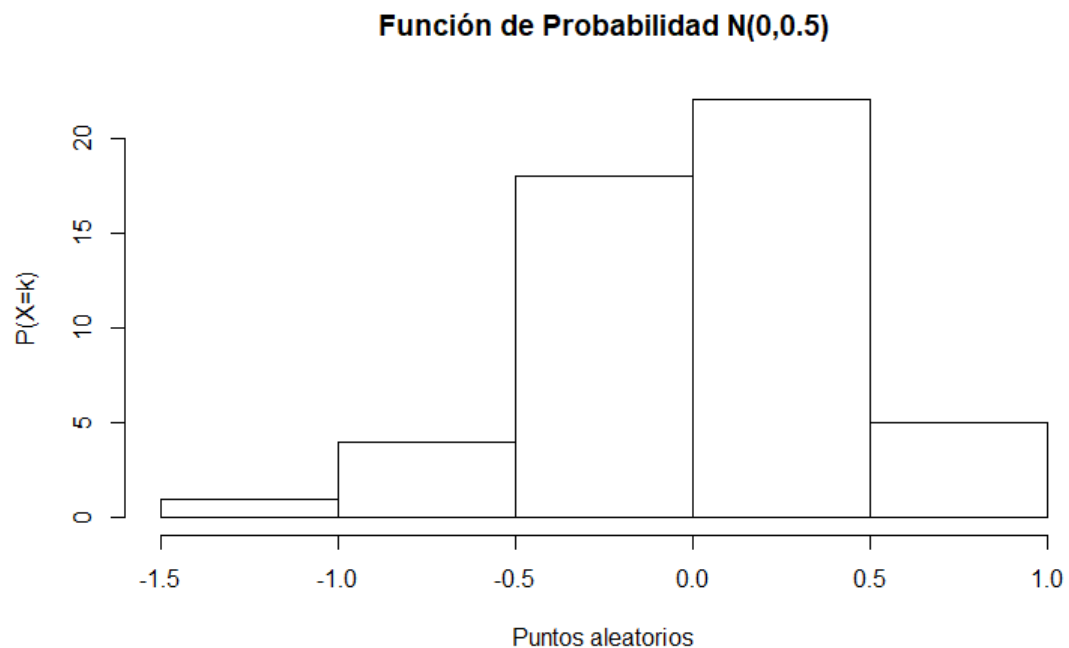
Para la distribución $N(0,1)$:

```
> hist(rnorm(50,0,1),xlab="Puntos aleatorios",ylab="P(X=k)",main="Función de Probabilidad N(0,1)")
```



Para la distribución $N(0,0.5)$:

```
> hist(rnorm(50,0,0.5),xlab="Puntos aleatorios",ylab="P(X=k)",main="Función de Probabilidad N(0,0.5)")
```



Para la distribución $N(0,2)$:

```
> hist(rnorm(50,0,2),xlab="Puntos aleatorios",ylab="P(X=k)",main="Función de Probabilidad N(0,2)")
```

