

# Programación y Uso de Librerías en R: Herramientas de Análisis y Visualización de Datos

Juan Luis Peñaloza Figueroa  
Universidad Complutense de Madrid

Milagros Dones Tacero  
Universidad Autónoma de Madrid

Carmen Gladys Vargas Pérez  
Universidad Complutense de Madrid

AÑO: 2025

## SCRIPT\_10: CAPÍTULO XII. DISTRIBUCIONES DE PROBABILIDAD EN R

### ## DISTRIBUCIÓN DE PROBABILIDAD BINOMIAL

```
> p<- 0,005
> P(X = 1) = ?
> choose(25,1)*p^1*(1-p)^(25-1)
> dbinom(1, size = 25, prob=0,005)
# Análogamente, se puede escribir como:
> sum(dbinom(0:6,8,0.4))
# Obtener simultáneamente varios percentiles
> qbinom(c(0.05,0.75,0.95),8,0.4)
# Simulamos 20 valores de la binomial
> rbinom(20,8,0.4)
# Gráfica función de masa o probabilidad
> plot(dbinom(0:10,8,0.4),type="h",xlab="k",ylab="P(X=k)",main="Función de Probabilidad B(8,0.4)")
# Gráfica de la función de distribución.
> plot(stepfun(0:10,pbinom(0:11,8,0.4)),xlab="k",ylab="F(k)",main="Función de distribución B(8,0.4)")
```

### ## DISTRIBUCIÓN PROBABILIDAD DE POISSON → $P(\lambda)$

```
> dpois(0:10, lambda=5, log=F)
# Representación gráfica de los valores del eje x
> x<-0:50
> lambda<-5
> plot(dpois(x,lambda), type="h", lwd=2,main="Función de masa de Poisson", ylab="P(X=x)", xlab="Número de eventos")
> lambda<-10
> lines(dpois(x,lambda), type="h", lwd=2,col=rgb(1,0,0,0.7))
> lambda<-20
> lines(dpois(x,lambda), type="h", lwd=2, col=rgb(0,1,0,0.7))
> ppois(5,lambda=10)
> sum(dpois(0:5, lambda=10))
```

### ## DISTRIBUCIÓN DE PROBABILIDAD GEOMÉTRICA

```
> dgeom(x=124,prob=0.01)
> pgeom(q=50,prob=0.01)
> qgeom(p=0.4,prob=0.01)
```

## **## DISTRIBUCIÓN DE PROBABILIDAD HIPERGEOMÉTRICA**

```
> M<-15
> N<-75
> n<-5
> k <- 3
> w <- choose(N,n)
> defectuoso <- choose(M,k)
> bueno <- choose(N-M,n-k)
> prob_a <- defectuoso* bueno/w
> prob_a
# Cuál es la probabilidad de seleccionar menos de tres mujeres
> M<-Mm
> éxito<-M
> fracaso<-N-M
> muestra<-n
> k<-2
> prob_m<-phyper(k,éxito,fracaso,muestra)
> prob_m
```

## **## DISTRIBUCIÓN DE PROBABILIDAD BINOMIAL NEGATIVA**

```
> r<-3
> p<-0.20
> n<-7-r
> dnbinom(x=n, size=r, prob=0.2)
# Calcular la media de una V.A. binomial negativa
> r<-3
> p<-0.2
> mean(rnbinom(n=10000, size=r,prob=p))+r
# Varianza de una V.A. binomial negativa
> r<-3
> p<-0.2
> var(rnbinom(n=100000, size=r,prob=p))
# Probabilidad acumulada de una binomial negativa
x<- 0:8
r<- 2
> cumsum(dnbinom(x,size=2,prob=1/2))
```

## **## DISTRIBUCIÓN JI-CUADRADO EN R**

```
> x<-0:15
> dchisq(x, df = 7)
# La función de densidad de un Ji.cuadrado
> curve(dchisq(x, df = 7), xlim = c(0,20), xlab = "Valores de
X",ylab="Densidad de Probabilidad")
# Información
> fob= fobservadas <- c(11, 32, 24) # Frecuencias observadas
> fes = fesperadas <- c(0.2, 0.5, 0.3)#Probabilidades esperadas (suman 1)
¿Son las probabilidades poblacionales iguales a 'p'?
> chisq.test(x = fob, p = fes)
```

## ## DISTRIBUCIONES DISCRETAS EMPÍRICAS

# Cargamos los datos en R

```
> url<-'https://raw.githubusercontent.com/fhernanb/datos/master/crab'
> crab<-read.table(file=url, header=T)
> head(crab)
> t1<-prop.table(table(crab$Sa))
> t1
> plot(t1,las=1,lwd=5,xlab="Número de satelites",ylab=
"Proporción")
```

## ## DISTRIBUCIONES DE PROBABILIDAD CONTINUAS

# Distribución de probabilidad normal  $\rightarrow n(\mu, \sigma)$

```
> pnorm(2.5)
> pnorm(40,mean=30,sd=4,lower.tail=TRUE)
> pnorm(21,mean=30,sd=4,lower.tail = FALSE)
> pnorm(35,mean=30,sd=4,lower.tail=TRUE)-pnorm(30,mean=30,
sd=4,lower.tail = TRUE)
```

# Distribución t-student

# Cálculo de la función de densidad de una t-student

```
> X=rnorm(10000, 170, 12)
> dt(x, df = 100)
> curve(dt(x, df = 100), xlim = c(-3,3), xlab = "Valores de
t", ylab = "Densidad de Probabilidad")
```

# Distribución f de Snedecor

# Cálculo de la densidad de un F de Snedecor

```
> df(x, df1 = 6, df2 = 3)
# Grafica de la densidad de una distribución F
> curve(df(x, df1 = 6, df2 = 3), xlim = c(0,10), xlab = "Valores
de F", ylab = "Densidad de Probabilidad")
> qf(0.07,df1=3,df2=10,lower.tail=F) # P [X > x]
> qf(0.93,df1=3,df2=10,lower.tail=T) # P[X ≤ x]
```

# Distribución beta

```
> pbeta(0.9, 4, 2, lower.tail = F)
> pbeta(0.5, 4, 2, lower.tail = T)
> qbeta(1/5, 4, 2, lower.tail = T)
```

# Función de densidad para una Beta(2,5).

```
> curve(dbeta(x, shape1=2, shape2=5), lwd=3, las=1,
ylab='Densidad')
```

# Distribución Weibull

```
> pweibull(0.75, 4.5, scale= 4^(1/4.5), lower.tail=F)
> qweibull(0.95, 4.5, scale = 4^(1/4.5), lower.tail = F)
```

# Distribución exponencial

```
> curve(dexp(x, rate=2.5), lwd=3, las=1, ylab='Densidad',
from=0, to=3)
```

```
pexp(q=1.5, rate=2.5) - pexp(q=0.5, rate=2.5)
```

# Calcular la función de distribución acumulada utilizando la ecuación acumulada

```
> prob.acum = 1 - e^(-lambda * x)
> prob.acum
> exp(q = x, rate = lambda)
```

