Soluciones a la hoja de ejercicios 3

Ejercicio 1

Para que $f(x)$ sea una función de densidad debe cumplirse que $\int_0^2 f(x)\,dx = 1$. Es decir

$$C\int_0^2 (4x - 2x^2)\,dx = 1$$

La integral es

$$C\left[2x^2 - \frac{2}{3}x^3\right]_0^2 = C\left[\frac{6x^2 - 2x^3}{3}\right]_0^2 = C\frac{8}{3} = 1$$

Por tanto

$$C = \frac{3}{8}$$

El valor esperado de X es

$$\begin{aligned} E(X) &= \frac{3}{8}\int_0^2 (4x^2 - 2x^3)\,dx \\ &= \frac{3}{8}\left[\frac{4x^3}{3} - \frac{x^4}{2}\right]_0^2 \\ &= 1 \end{aligned}$$

Para obtener la probabilidad pedida podemos comenzar definiendo la función de distribución

$$F(x) = \frac{3}{8}\int_0^x (4t - 2t^2)\,dt = \frac{3}{8}\left[2t^2 - \frac{2}{3}t^3\right]_0^x = \frac{3x^2 - x^3}{4}$$

Entonces

$$P(X > 1) = 1 - F(1) = \frac{1}{2}$$

La representación gráfica de $f(x)$ aparece en la figura 1. El código es

```
x <- seq(0, 2, 0.001)
fx<- 3*(4*x - 2*x^2)/8
plot(x, fx, col="darkolivegreen4", lwd=1.5, type="l", xlab="X", ylab="f(x)")
```

El valor esperado y la probabilidad requerida pueden calcularse del siguiente modo en R

```
f1 <- function(x){3*(4*x^2 - 2*x^3)/8}
f2 <- function(x){3*(4*x - 2*x^2)/8}
Ex <- integrate(f1, lower = 0, upper = 2)
Pr <- integrate(f2, lower = 1, upper = 2)
cat(sprintf("E(X) = %5.3f, Pr = %5.3f\n", Ex$value, Pr$value))
```

```
## E(X) = 1.000, Pr = 0.500
```

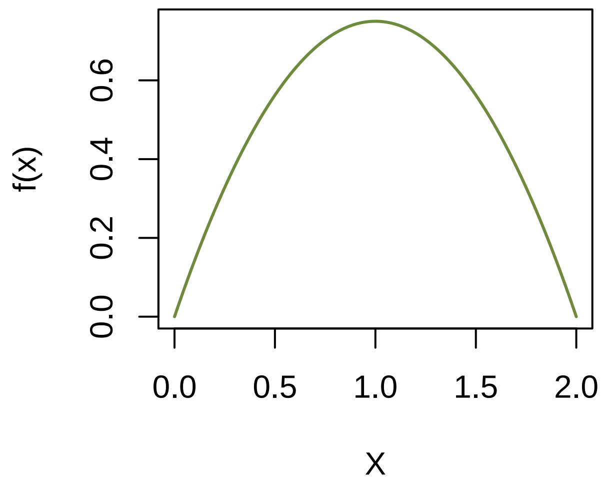

Figura 1: Función de densidad

Ejercicio 2

El código R es

```
x <- 0:12
f <- dpois(x,12)
plot(x,f, type="h")
```

```
C5 <- qpois(0.05, 12)
C95 <- qpois(0.95, 12)

Pm15 <- ppois(15,12,lower.tail=F)
Pm15_condicionada <- ppois(15,12,lower.tail=F)/ppois(10,12,lower.tail=F)

cat(sprintf("Centil 5: %1.0f, Centil 95: %1.0f, P(X>15)= %5.3f, P(X>15|X>10)=%5.3f",
            C5, C95, Pm15, Pm15_condicionada))
```

```
## Centil 5: 7, Centil 95: 18, P(X>15)= 0.156, P(X>15|X>10)=0.238
```

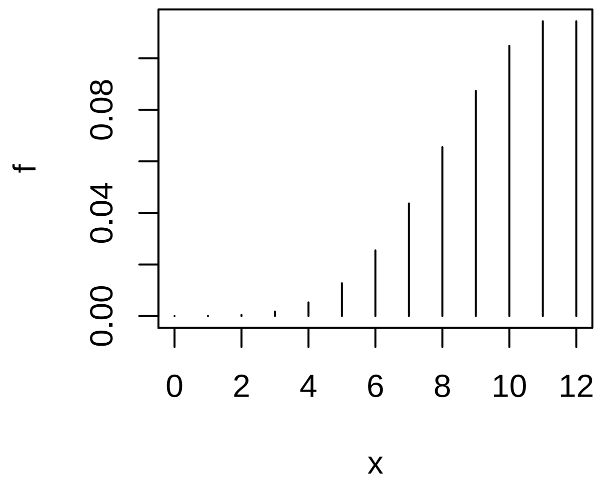

Figura 2: Función de probabilidad Poisson (12)

Ejercicio 3

El siguiente código realiza la simulación indicada.

```
a_True <- 5
b_True <- 2

alpha <- function(m,v) m *(m*(1-m)/v - 1)
beta <- function(m,v) (1-m)*(m*(1-m)/v - 1)

n <- 100
muestras <- 200

a_Estimada <- rep(0, muestras)
b_Estimada <- rep(0, muestras)

for(muestra in 1:muestras){
    x <- rbeta(n, a_True, b_True)

    media <- mean(x)
    varianza <- var(x)

    a_Estimada[muestra] <- alpha(media, varianza)
    b_Estimada[muestra] <- beta(media, varianza)
}

media_a <- mean(a_Estimada)
media_b <- mean(b_Estimada)
se_a <- sd(a_Estimada)
```

```r
se_b <- sd(b_Estimada)
correlacion <- cor(a_Estimada,b_Estimada)


cat(sprintf(
"N = %1.0f, media alpha = %6.4f, Se alpha = %6.4f, media beta = %6.4f, Se beta = %6.4f, r = %5.3f",
n, media_a, se_a, media_b, se_b, correlacion))
```

```
## N = 100, media alpha = 5.2041, Se alpha = 0.8397, media beta = 2.0669, Se beta = 0.3120, r = 0.863
```

```r
par(mfrow=c(1,2))
hist(a_Estimada, main="alpha")
hist(b_Estimada, main="beta")
```

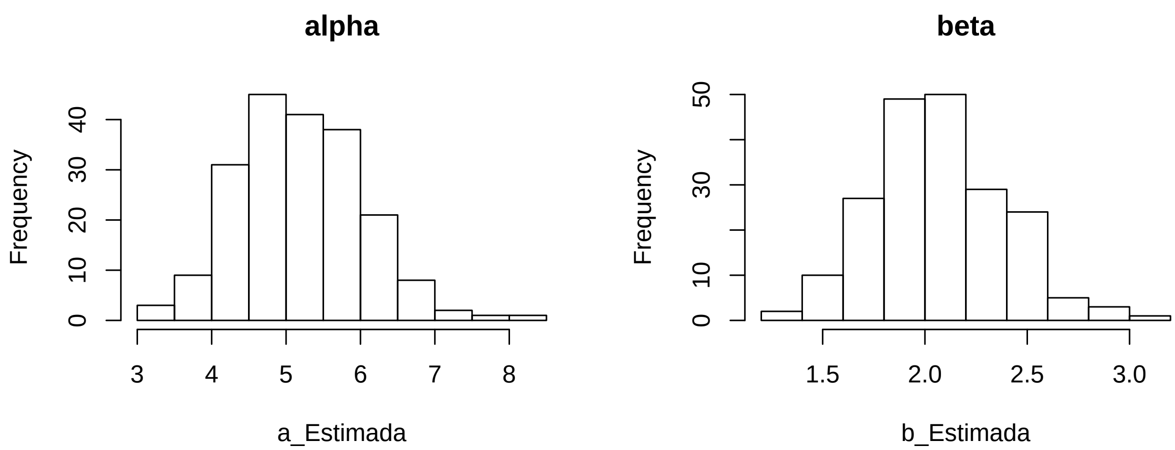

Figura 3: Recuperación de parámetros