

Examen 1 2024

Javier Orín

2025-12-09

#Problema 1 (3 puntos). Cierta ceca fabrica monedas de forma que la probabilidad de que salga cara si tiramos al aire la moneda i -ésima es p_i . Es decir, distintas monedas tienen distinta probabilidad de cara. Cada p_i puede estimarse con mucha precisión, por lo que dada una muestra concreta de n monedas sus p_i se suponen conocidas ($i = 1, \dots, n$). Las p_i provienen de una población beta $B(q, q)$, es decir, con ambos parámetros iguales. Se dispone de una realización muestral de tamaño $n = 10$, a saber, $(p_1, \dots, p_{10}) = (0.449, 0.515, 0.432, 0.526, 0.433, 0.539, 0.560, 0.546, 0.476, 0.630)$. ##a) Halla un estadístico suficiente minimal para q (0,6 puntos). Sabemos que un estadístico es suficiente minimal si y solo si induce la siguiente partición:

$$\vec{x} \sim \vec{y} \iff \frac{f(\vec{x}; q)}{f(\vec{y}; q)} \text{ no depende de } q$$

Veamos cómo es la función de densidad de la variable aleatoria:

$$f_X(x) = \frac{1}{\beta(q, q)} (x - x^2)^{q-1} \cdot I_{(0,1)}(x)$$

Pasemos a comparar ahora dos m.a.s. de tamaño n de la variable aleatoria:

$$\frac{f(\vec{x}; q)}{f(\vec{y}; q)} = \prod_{i=1}^n \left(\frac{x_i - x_i^2}{y_i - y_i^2} \right)^{q-1} = \left(\prod_{i=1}^n \frac{x_i - x_i^2}{y_i - y_i^2} \right)^{q-1}$$

Tenemos que el cociente de densidades es una base elevado al exponente $q - 1$. La única forma de que el resultado no dependa de q es hacer que la base sea igual a 1; o, equivalentemente, considerar el siguiente estadístico:

$$T(\vec{x}) = \prod_{i=1}^n x_i - x_i^2$$

Otra posible alternativa es expresar la variable aleatoria como parametrización natural de la familia exponencial. Se obtiene una transformación biyectiva de esta solución por medio de logaritmos.

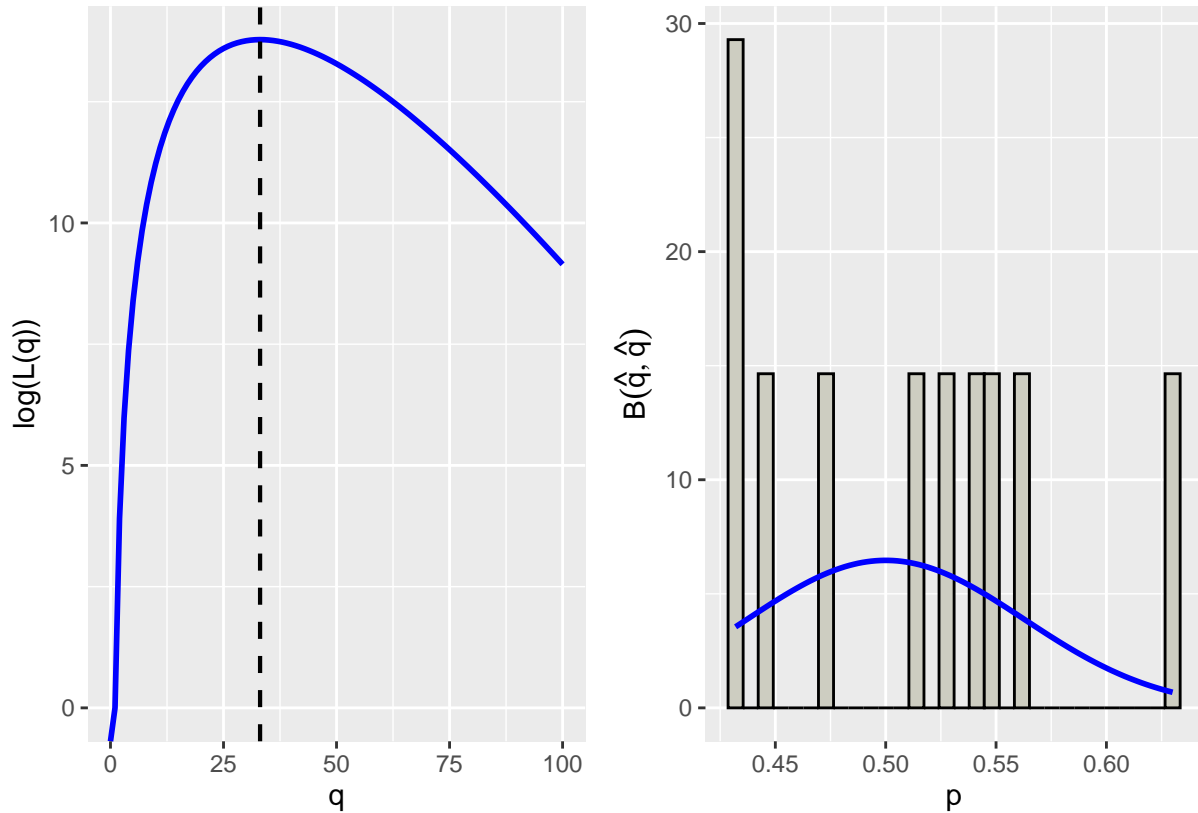
##b) Halla una estimación de q por el método de máxima verosimilitud (0,8 puntos). Cargamos la muestra en R:

```
prob_caras = c(0.449, 0.515, 0.432, 0.526, 0.433, 0.539, 0.560, 0.546, 0.476, 0.630)
```

Definimos la función `log_verosimilitud` para optimizar numéricamente:

```
log_verosimilitud = function(q) {  
  n = length(prob_caras)  
  s = sum(log(prob_caras) + log(1 - prob_caras))  
  (q-1)*s - n*log(beta(q,q))  
}  
(q_emv = optimise(f=log_verosimilitud, interval = c(0, 100), maximum=TRUE)$maximum)
```

```
## [1] 33.08617
```



c) Halla una estimación de q por el método de los momentos (0,8 puntos).

Por ser $X \sim B(q, q)$, sabemos

$$E(X) = \frac{q}{q+q} = \frac{1}{2} \qquad \text{Var}(X) = \frac{q^2}{(2q)^2 \cdot (2q+1)} = \frac{1}{8q+4}$$

El método de los momentos aproxima los momentos poblacionales por los muestrales. Por tanto, despejamos q como sigue:

$$s^2 = \frac{1}{8q+4} \Rightarrow 8q+4 = \frac{1}{s^2} \Rightarrow q = \frac{1}{8} \left(\frac{1}{s^2} - 4 \right) = \frac{1}{8s^2} - \frac{1}{2}$$

Veamos qué valor toma para la muestra:

```
(q_mom = -0.5 + 1/(8 * var(prob_caras)))
```

```
## [1] 30.57357
```

Es cercano al valor obtenido por el EMV.

d) Halla un intervalo de confianza del 95% para q .

Debido a la alta simetría de la función de densidad, podemos usar el TCL aun teniendo un tamaño muestral bajo. En este caso, tenemos que despejar q de la varianza en lugar de la muestra o trabajar con una $N(0,1)^2 \equiv \chi_1^2$.

#Problema 2 (3 puntos). Se quiere estudiar el tiempo de vida (en años) de los dispositivos eléctricos de la marca Chispazo. Para ello se estudia una muestra de 100 de estos dispositivos electricos, recogiendo los tiempos de vida resultantes en el conjunto de datos `Chispazo.RData`. ##a) Basandote en los datos recogidos, estudia desde un punto de vista descriptivo si la distribución del tiempo de vida sigue aproximadamente una distribucion de Weibull $W(\alpha, \lambda)$ con parametro de forma $\alpha = 2$. Para ello, estima puntualmente el parametro de escala λ de la distribucion haciendo uso de su estimador maximo-verosímil. La densidad de una variable aleatoria Weibull $X \sim W(\alpha, \lambda)$ viene dada por:

$$f_X(x) = \frac{\alpha}{\lambda} \cdot \left(\frac{x}{\lambda}\right)^{\alpha-1} \cdot \exp\left(-\left(\frac{x}{\lambda}\right)^\alpha\right) \cdot I_{(0,+\infty)}(x) \stackrel{\alpha=2}{=} \frac{2x}{\lambda^2} \cdot \exp\left(\frac{-x^2}{\lambda^2}\right) \cdot I_{(0,+\infty)}(x)$$

De este modo, sea $\vec{x}_n = (x_1, \dots, x_n)$ una m.a.s. extraída de X , su log-verosimilitud viene dada por:

$$\log L(\lambda; \vec{x}_n) = \sum_{i=1}^n \left[\log(2x_i) - 2 \log(\lambda) - \frac{x_i^2}{\lambda^2} \right] = -2n \log(\lambda) + \sum_{i=1}^n \left[\log(2x_i) - \frac{x_i^2}{\lambda^2} \right]$$

Derivamos respecto del parámetro desconocido para hallar el EMV:

$$\frac{\partial}{\partial \lambda} \log L(\lambda; \vec{x}_n) = -\frac{2n}{\lambda} + \frac{2 \sum_{i=1}^n x_i^2}{\lambda^3} = 0 \iff \frac{n}{\lambda} = \frac{\sum_{i=1}^n x_i^2}{\lambda^3} \iff \lambda \left(n\lambda^2 - \sum_{i=1}^n x_i^2 \right) = 0$$

Como el parámetro λ ha de estar en $(0, +\infty)$, esto equivale a su vez a exigir:

$$n\lambda^2 = \sum_{i=1}^n x_i^2 \Rightarrow \lambda_{MV} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}$$

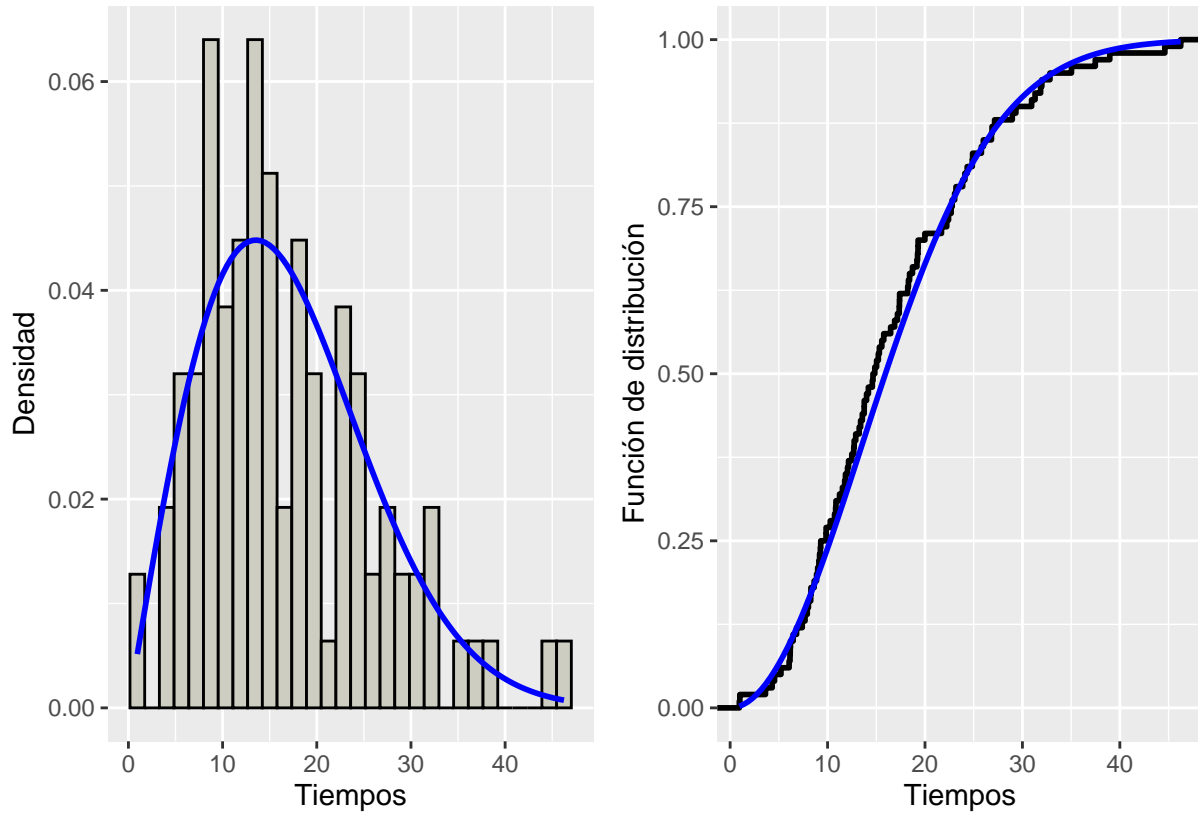
Para ver que es un máximo, necesitamos analizar el signo de la segunda derivada:

$$\frac{\partial^2}{\partial \lambda^2} \log L(\lambda_{MV}; \vec{x}_n) = \frac{2n}{\lambda_{MV}^2} - \frac{6 \sum_{i=1}^n x_i^2}{\lambda_{MV}^4} = \frac{2n^2}{\sum_{i=1}^n x_i^2} - \frac{6n^2}{\sum_{i=1}^n x_i^2} < 0$$

Es en efecto un máximo. Veamos qué valor toma para la muestra:

```
load("Chispazo.RData")
est_lambda_mv = function(muestra) {
  sqrt(mean(muestra^2))
}
(lambda_mv = est_lambda_mv(Chispazo))
```

```
## [1] 19.14119
```



Gráficamente, se ve una clara semejanza. Veamos si se cumple también para un análisis cuantitativo.

Simulemos muchas muestras del mismo tamaño y recopilemos medidas de error para ellas:

```
nr = 1e3
n = length(Chispazo)
results = data.frame()
for (rep in 1:nr) {
  muestra = rweibull(n=n, shape=2, scale=lambda_mv)
  error = sort(Chispazo) - sort(muestra)
  results[rep, "sesgo"] = mean(error)
  results[rep, "error_absoluto"] = mean(sapply(error, abs))
  results[rep, "ECM"] = mean(sapply(error, function(x) x^2))
}
summary(results)
```

| ## | sesgo | error_absoluto | ECM |
|------------|----------|----------------|-----------------|
| ## Min. | :-2.9168 | Min. :0.4842 | Min. : 0.4563 |
| ## 1st Qu. | :-0.8614 | 1st Qu.:1.0319 | 1st Qu.: 2.0707 |
| ## Median | :-0.2722 | Median :1.2510 | Median : 2.9686 |
| ## Mean | :-0.2509 | Mean :1.3339 | Mean : 3.3809 |
| ## 3rd Qu. | :0.4069 | 3rd Qu.:1.5657 | 3rd Qu.: 4.2309 |
| ## Max. | : 2.3695 | Max. :2.9460 | Max. :12.6038 |

Se observan errores muy bajos.