

Proyecto 2

Sara Leiva, Pedro Pablo Sanín

Introducción

La distribución Gamma con parámetros α y β se usa en las ciencias para modelar tiempos de vida de organismos o artefactos. Su densidad de probabilidad está dada por

$$\frac{x^{\alpha-1}e^{-x/\beta}}{\Gamma(\alpha)\beta^\alpha}$$

y está soportada para $x > 0$. Se tiene además que

$$\mu = \mathbb{E}(X) = \alpha\beta, \quad \sigma^2 = \text{Var}(X) = \alpha\beta^2$$

.

El objetivo de este proyecto es calcular el Estimador de Máxima Verosimilitud del parámetro $\theta = (\alpha, \beta)$ de este modelo, y verificar la validez del conjunto elíptico de confianza obtenido al estimar el parámetro con el Teorema del Límite Central para Estimadores de Máxima Verosimilitud biparamétrico.

A lo largo del proyecto, supondremos que contamos con n muestras iid. $\mathcal{X} = (X_1, \dots, X_n)$ con distribución Gamma(α, β) para $\alpha = \alpha_0 = 2.5$ y $\beta = \beta_0 = 4.6$.

Cálculo numérico del Estimador de Máxima Verosimilitud para el parámetro θ

Método de Newton-Raphson para calcular el EMV La verosimilitud del parámetro θ para la muestra \mathcal{X} está dada por la función bivariada

$$L(\theta) = \prod_{i=1}^n f(X_i; \alpha, \beta) = \prod_{i=1}^n \frac{X_i^{\alpha-1} e^{-X_i/\beta}}{\Gamma(\alpha)\beta^\alpha}$$

. Así, la log-verosimilitud es:

$$l(\theta) = \sum_{i=1}^n \left((\alpha - 1) \ln(X_i) - \frac{X_i}{\beta} - \ln(\Gamma(\alpha)) - \alpha \ln(\beta) \right) = -n \ln(\Gamma(\alpha)) - n\alpha \ln(\beta) + \sum_{i=1}^n \left((\alpha - 1) \ln(X_i) - \frac{X_i}{\beta} \right).$$

Para encontrar el valor de θ que minimiza $l(\theta)$, es decir, el estimador de máxima verosimilitud para los valores observados, debemos solucionar las ecuaciones

$$\frac{\partial l(\theta)}{\partial \alpha} = 0, \quad \frac{\partial l(\theta)}{\partial \beta} = 0.$$

Vea entonces que

$$\frac{\partial l(\theta)}{\partial \alpha} = -n \frac{\Gamma'(\alpha)}{\Gamma(\alpha)} - n \ln(\beta) + \sum_{i=1}^n \ln(X_i).$$

Asimismo

$$\frac{\partial l(\theta)}{\partial \beta} = -\frac{n\alpha}{\beta} + \frac{1}{\beta^2} \sum_{i=1}^n X_i.$$

Para solucionar el sistema de ecuaciones, podemos usar métodos numéricos, pues la solución simbólica es difícil. En este caso, usaremos el método de Newton-Raphson bivariado. Para esto, debemos identificar un estimador inicial $\tilde{\theta}_0$ y a partir de esto construimos las aproximaciones a la raíz por la fórmula:

$$\tilde{\theta}_{k+1} = \tilde{\theta}_k - [Df(\tilde{\theta}_k)]^{-1} f(\tilde{\theta}_k),$$

siendo $f : X \subseteq \mathbb{R}^2 \longrightarrow \mathbb{R}^2$ con

$$f(\alpha, \beta) = \left(\frac{\partial l(\theta)}{\partial \alpha}(\alpha, \beta), \frac{\partial l(\theta)}{\partial \beta}(\alpha, \beta) \right)$$

Así, si $\tilde{\theta}_k = (\tilde{\alpha}_k, \tilde{\beta}_k)$, tenemos que

$$Df(\tilde{\theta}_k) = \begin{pmatrix} -n \frac{\Gamma''(\tilde{\alpha}_k)\Gamma(\tilde{\alpha}_k) - \Gamma'(\tilde{\alpha}_k)^2}{\Gamma(\tilde{\alpha}_k)^2} & -\frac{n}{\tilde{\beta}_k} \\ -\frac{n}{\tilde{\beta}_k} & \frac{n\tilde{\alpha}_k}{\tilde{\beta}_k^2} - 2\frac{\sum_{i=1}^n X_i}{\tilde{\beta}_k^3} \end{pmatrix}.$$

Invirtiendo la matriz, obtenemos:

$$[Df(\tilde{\theta}_k)]^{-1} = \frac{1}{-n \frac{\Gamma''(\tilde{\alpha}_k)\Gamma(\tilde{\alpha}_k) - \Gamma'(\tilde{\alpha}_k)^2}{\Gamma(\tilde{\alpha}_k)^2} \left(\frac{n\tilde{\alpha}_k}{\tilde{\beta}_k^2} - 2\frac{\sum_{i=1}^n X_i}{\tilde{\beta}_k^3} \right) - \frac{n^2}{\tilde{\beta}_k^2}} \begin{pmatrix} \frac{n\tilde{\alpha}_k}{\tilde{\beta}_k^2} - 2\frac{\sum_{i=1}^n X_i}{\tilde{\beta}_k^3} & \frac{n}{\tilde{\beta}_k} \\ \frac{n}{\tilde{\beta}_k} & -n \frac{\Gamma''(\tilde{\alpha}_k)\Gamma(\tilde{\alpha}_k) - \Gamma'(\tilde{\alpha}_k)^2}{\Gamma(\tilde{\alpha}_k)^2} \end{pmatrix}.$$

Sabemos que, para asegurarnos de que el método de Newton-Raphson converge, debemos elegir un estimador inicial apropiado. Sabemos que podemos encontrar estimadores suficientemente buenos para α y β usando los momentos de la distribución. Sabemos de las ecuaciones anteriores que

$$\alpha = \frac{\mu^2}{\sigma^2}, \quad \beta = \frac{\sigma^2}{\mu}$$

y así, podemos definir

$$\tilde{\alpha}_0 = \frac{\overline{X}^2}{s^2}, \quad \tilde{\beta}_0 = \frac{s^2}{\overline{X}}$$

Calculando el EMV A continuación mostramos el código R usado para calcular el estimador

```
set.seed(1)

mle_gamma_nr <- function(sample, precision = 1e-6, max_iterations = 50){
  # Estadísticos básicos
  sample.size <- length(sample)
  sample.mean <- mean(sample)
  sample.variance <- var(sample)
  sample.sum <- sum(sample)
  sample.sumlog <- sum(log(sample))

  iterations <- 0L
  diferencia <- Inf

  # Gradiente de la log-verosimilitud
  likelihood_gradient <- function(theta_k){
    alpha_k <- theta_k[1]
```

```

    beta_k <- theta_k[2]
    dalpha <- -sample.size * digamma(alpha_k) - sample.size * log(beta_k) + sample.sumlog
    dbeta <- (-sample.size * alpha_k / beta_k) + (sample.sum / beta_k^2)
    return(c(dalpha, dbeta))
}

# Inversa del Hessiano
inverse_hessian <- function(theta_k){
  alpha_k <- theta_k[1]
  beta_k <- theta_k[2]
  a <- -sample.size * trigamma(alpha_k)
  b <- -sample.size / beta_k
  d <- (sample.size * alpha_k / beta_k^2) - 2 * (sample.sum / beta_k^3)
  detA <- a * d - b^2
  return((1 / detA) * matrix(c(d, -b, -b, a), nrow = 2, byrow = TRUE))
}

# Inicialización por momentos
theta_prev <- c((sample.mean)^2 / sample.variance, sample.variance / sample.mean)

# Verificamos que los estimadores estén bien definidos y sean positivos.
if (any(!is.finite(theta_prev)) || any(theta_prev <= 0)) {
  # theta_prev <- pmax(theta_prev, 1e-8)
}

repeat {
  # Paso de Newton: theta_new = theta_prev - inv(H) %% grad
  paso <- drop(inverse_hessian(theta_prev) %% likelihood_gradient(theta_prev))
  theta_new <- theta_prev - paso

  theta_new[!is.finite(theta_new) | theta_new <= 0] <- 1e-8

  iterations <- iterations + 1L
  diferencia <- max(abs(theta_new - theta_prev))

  if (diferencia < precision || iterations >= max_iterations) break
  theta_prev <- theta_new
}

return(list(
  theta = theta_new,
  iteraciones = iterations,
  convergio = (diferencia < precision)
))
}

```

Ahora, generaremos 1000 muestras de tamaños $n = 100, 200, 500, 1000$ y 2000 , y dibujaremos para comparar los distintos estimadores obtenidos para cada n .

