MSMS 106 : Practical 17

Ananda Biswas

December 18, 2024

➜ Objective

To find Maximum Likelihood Estimate of the parameters of the Gamma Distribution given as follows :

$$f_X(x) = \frac{1}{\Gamma(\alpha)\beta^\alpha}\, e^{-\frac{x}{\beta}}\, x^{\alpha-1}\, I_{(0,\infty)}(x);\ \alpha > 0,\ \beta > 0$$

and to compare ML estimates for different sample sizes.

➜ Theory

α is called the shape parameter and β is called the scale parameter of the distribution.

For a sample of size n, the likelihood function is

$$L(\alpha, \beta) = \left(\frac{1}{\Gamma(\alpha)\beta^\alpha}\right)^n \cdot exp\left\{-\frac{1}{\beta}\sum_{i=1}^n x_i\right\} \left(\prod_{i=1}^n x_i\right)^{\alpha-1}.$$

The log-likelihood function is

$$l(\alpha, \beta) = -n\ln(\Gamma(\alpha)) - \alpha n \ln(\beta) - \frac{1}{\beta}\sum_{i=1}^n x_i + (\alpha - 1)\sum_{i=1}^n \ln x_i.$$

The partial derivative of the log-likelihood w.r.t. β is

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

Setting it to 0 and solving for β we get,

$$\hat{\beta} = \frac{1}{\alpha n}\sum_{i=1}^n x_i = \frac{\bar{x}}{\alpha}. \tag{1}$$

The partial derivative of the log-likelihood w.r.t. α is

$$\begin{aligned}\frac{\partial}{\partial\alpha} l(\alpha, \beta) &= -n\frac{\Gamma'(\alpha)}{\Gamma(\alpha)} - n\ln(\beta) + \sum_{i=1}^n \ln x_i \\ &= -n\,\psi(\alpha) - n\ln\left(\frac{\bar{x}}{\alpha}\right) + \sum_{i=1}^n \ln x_i \\ &= -n\,\psi(\alpha) - n\ln(\bar{x}) + n\ln(\alpha) + \sum_{i=1}^n \ln x_i.\end{aligned}$$

$\psi(z) = \dfrac{d}{dz}\ln(z) = \dfrac{\Gamma'(z)}{\Gamma(z)}$ is called **digamma function**.

Setting the partial derivative w.r.t. α to 0, we get an equation in α given by

$$g(\alpha) = -n\,\psi(\alpha) - n\ln(\bar{x}) + n\ln(\alpha) + \sum_{i=1}^{n} \ln x_i = 0. \tag{2}$$

We cannot obtain any closed-form solution of $g(\alpha)$, so we opt for numerical solution.

$$g'(\alpha) = -n\,\psi'(\alpha) + \frac{n}{\alpha}. \tag{3}$$

$\psi'(\alpha)$ is called **trigamma function**.

Using (2) and (3), we get an approximate solution of $g(\alpha)$ by **Newton-Raphson method**. That solution is indeed ML estimate of α. By using that estimate of α in (1), we will obtain ML estimate of β.

- To compare MLEs from different sample sizes, we compare their MSEs.

For a fixed sample size n, $\text{MSE}(\hat{\theta}_{\text{MLE}}) = \frac{1}{k}\sum_{i=1}^{k}(\hat{\theta}_i - \theta_0)^2$, where $\hat{\theta}_1$, $\hat{\theta}_2$, $\hat{\theta}_3$, $\ldots$, $\hat{\theta}_k$ are MLEs from different samples of fixed size n and θ_0 is the true value of θ.

R Program

The following function takes random sample, initial approximation of α and number of iteration as inputs and gives $\hat{\alpha}$ and $\hat{\beta}$ as outputs.

```r
Gamma_MLE <- function(gamma_sample, shape_initial, n_iteration){

  a <- c(shape_initial)

  n <- length(gamma_sample)

  f1 <- function(alpha){

    result <- - n * digamma(alpha) -
                n * log(mean(gamma_sample)) +
                n * log(alpha) +
                sum(log(gamma_sample))

    return(result)
  }

  f2 <- function(alpha){
    return(-n * trigamma(alpha) + n / alpha)
  }

  iterations <- n_iteration
  for (i in 2:iterations) {
    a[i] <- a[i-1] - f1(a[i-1]) / f2(a[i-1])

    if(abs(f1(a[length(a)])) < 0.001) break
  }

  alpha_hat <- a[length(a)]

  beta_hat <- mean(gamma_sample) / alpha_hat

  return(c(alpha_hat, beta_hat))
}
```

Now we calculate MLEs for different sample sizes.

```
n <- c(100, 200, 500, 1000, 5000, 10000, 100000)
```

```
estimated_shape <- c(); estimated_scale <- c()
```

```
for (i in 1:length(n)) {
  our_sample <- rgamma(n[i], shape = 6, scale = 2)
  temp <- Gamma_MLE(our_sample, shape_initial = 8, n_iteration = 1000)
  estimated_shape[i] <- temp[1]
  estimated_scale[i] <- temp[2]
}
```

```
Gamma_MLE_df1 <- data.frame(sample_size = n,
                            shape_hat = estimated_shape,
                            scale_hat = estimated_scale)
```

```
Gamma_MLE_df1

##   sample_size shape_hat scale_hat
## 1       1e+02  6.840379  1.807420
## 2       2e+02  5.316375  2.162303
## 3       5e+02  6.432899  1.868096
## 4       1e+03  5.701569  2.106921
## 5       5e+03  5.920148  2.028113
## 6       1e+04  6.046408  1.991143
## 7       1e+05  6.027432  1.989229
```

✎ As sample size increases, the estimates of parameters seem to converge at 6 and 2 respectively.

Now we shall compare the MSEs.

```
MSE_alpha_hat <- c(); MSE_beta_hat <- c()
```

```
for(j in 1:length(n)){

  alpha_hats <- c(); beta_hats <- c()

  for (i in 1:100) {
    a_sample <- rgamma(n[j], shape = 6, scale = 2)
    temp <- Gamma_MLE(a_sample, shape_initial = 8, n_iteration = 1000)
    alpha_hats[i] <- temp[1]
    beta_hats[i] <- temp[2]
  }

  MSE_alpha_hat[j] <- mean( (alpha_hats - 6)^2 )

  MSE_beta_hat[j] <- mean( (beta_hats - 2)^2 )
}
```

```
Gamma_MLE_df2 <- data.frame(sample_size = n,
                            MSE_alpha_hat = MSE_alpha_hat,
                            MSE_beta_hat = MSE_beta_hat)
```

```
Gamma_MLE_df2

##   sample_size MSE_alpha_hat MSE_beta_hat
## 1       1e+02  0.7976293640 7.736378e-02
## 2       2e+02  0.3385549703 4.205408e-02
## 3       5e+02  0.1408441806 1.517926e-02
## 4       1e+03  0.0641654296 7.546583e-03
## 5       5e+03  0.0172408067 2.109638e-03
## 6       1e+04  0.0077781313 8.895092e-04
## 7       1e+05  0.0006363066 7.703907e-05
```

Conclusion

✎ As sample size increases, MSEs of both the parameters decrease monotonically. This implies MLEs give better estimates as sample size increases. For larger and larger samples, the MLEs will smoothly converge to the true values of the parameters respectively.