Tarea 11

Simulación estocástica

Marco Antonio Andrade Barrera

19 de abril de 2018

Queremos generar $Ga(x|\alpha, \beta)$, $\alpha > 1$, $q(y) = Ga(y|a, 1)$.

Algoritmo:

1. $y = -\sum_{j=1}^{a} u_j \; u_j \sim U(u|0, 1)$
2. $\alpha(x_n, y) = min\{1, (y/x_n)^{\alpha-a} \cdot exp(-(\beta - 1)(y - x_n))\}$
3. Generar $u \sim U(u|0m1)$. Si $u \leq \alpha(x_n, y)$ entonces hacer $x_{n+1} = y$ si no hacer $x_{n+1} = x_n$.

Implementar el algoritmo para $(\alpha, \beta) = (5.9, 2.145)$.

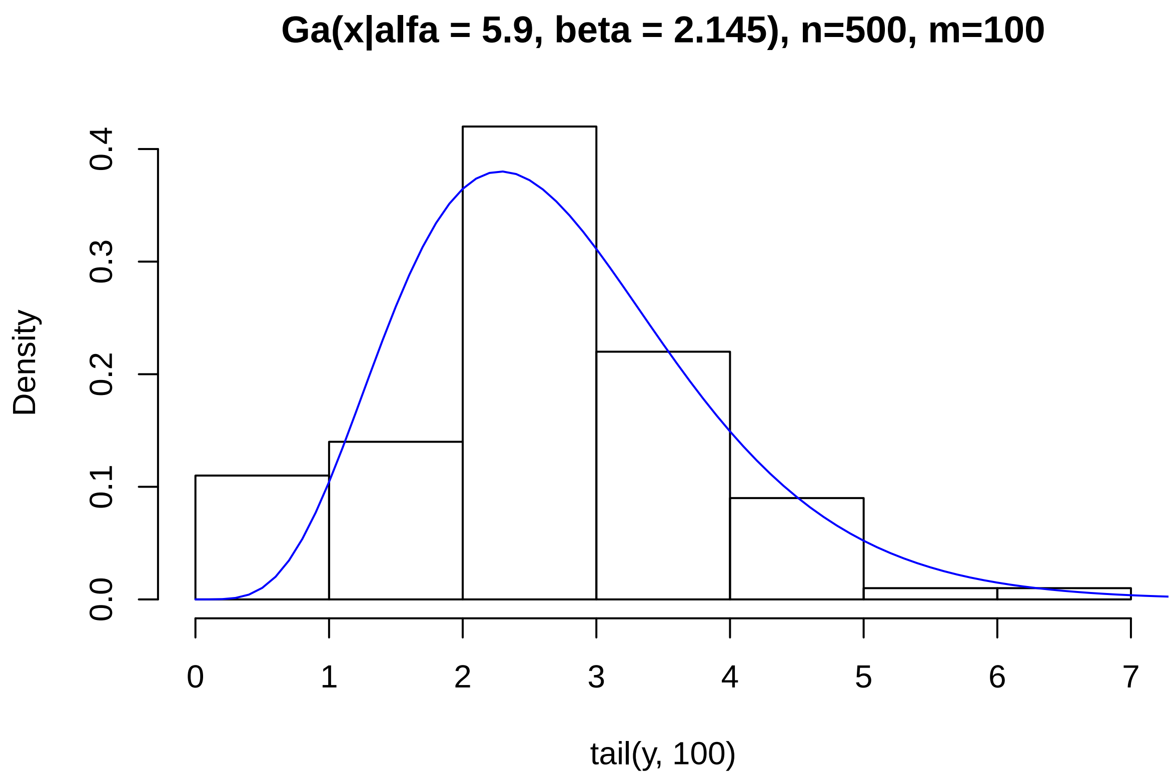
- Correr la cadena para $n \in \{500, 1000, 5000\}$
- Generar $m = 100$ extras
- Estimar el cuantil 0.37 y la media y comparar con los valores verdaderos.
- Usar las $n + m$ observaciones y verificar la convergencia del cuantil 0.37 y la media (de 10 en 10).

En se siguiente bloque de código se implementa una función, usando el algoritmo anterior, para generar una cadena de n valores.

```
alfa = 5.9
beta = 2.145
#x0 = 1 es el valor inicial
algo <- function(x0 = 1,n=600,alfa = 5.9, beta = 2.145){
  x <- list()
  x[1] <- x0
  a <- floor(alfa)
  for(i in 2:n){
    uj <- runif(n = a)
    y <- - sum(log(uj))
    xn <- as.numeric(x[i-1])
    alp <- min(1, (y/xn)^(alfa - a) * exp(-(beta - 1) * (y-xn)) )
    u <- runif(n = 1)
    if(u <= alp){
      x[i] <- y
    } else {
        x[i] <- x[i-1]
    }
  }
  unlist(x)
}
```

Para $n = 500$ y $m = 100$ extras tenemos:

```r
set.seed(151)
y <- algo(x0 = 1, alfa, beta, n = 500+100)
hist(tail(y,100),freq = F, main = "Ga(x|alfa = 5.9, beta = 2.145), n=500, m=100")
  curve(dgamma(x,shape=5.9,rate=2.145),0,10,add = T,col="blue")
```




```r
#Cuantil 0.37 real
qgamma(0.37,shape = alfa,rate = beta)
```

```
## [1] 2.250377
```

```r
#Cuantil empírico
quantile(tail(y,100),0.37)
```

```
##      37%
## 2.303314
```

```r
#Media real
alfa/beta
```

```
## [1] 2.750583
```

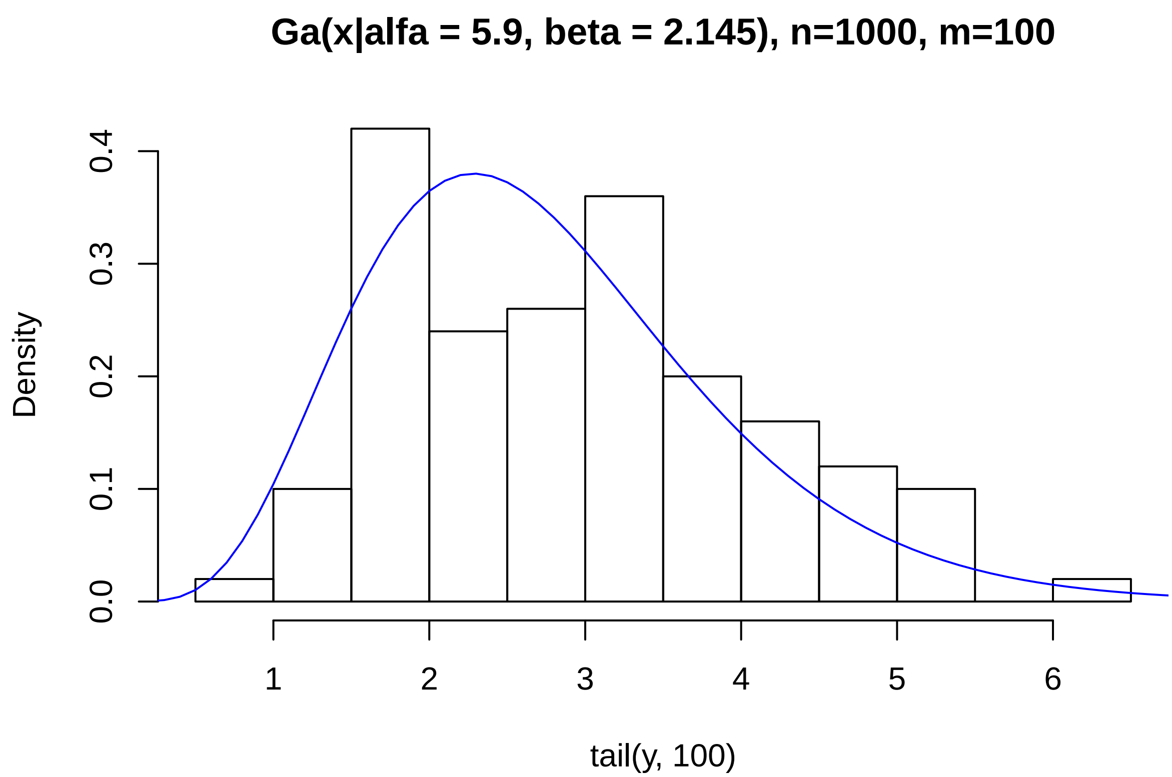
```r
#Media empírica
mean(tail(y,100))
```

```
## [1] 2.638111
```

En ambos casos, la diferencia entre el valor real y el empírico es de alrededor de 0.1, lo cual se traduce en un error de aproximadamente 5%.

Para $n = 1000$ y $m = 100$ extras tenemos:

```
set.seed(151)
y <- algo(x0 = 1, alfa, beta, n = 1000+100)
hist(tail(y,100),freq = F, main = "Ga(x|alfa = 5.9, beta = 2.145), n=1000, m=100")
  curve(dgamma(x,shape=5.9,rate=2.145),0,10,add = T,col="blue")
```




```
#Cuantil 0.37 real
qgamma(0.37,shape = alfa,rate = beta)
```

```
## [1] 2.250377
```

```
#Cuantil empírico
quantile(tail(y,100),0.37)
```

```
##      37%
## 2.143778
```

```
#Media real
alfa/beta
```

```
## [1] 2.750583
```

```
#Media empírica
mean(tail(y,100))
```

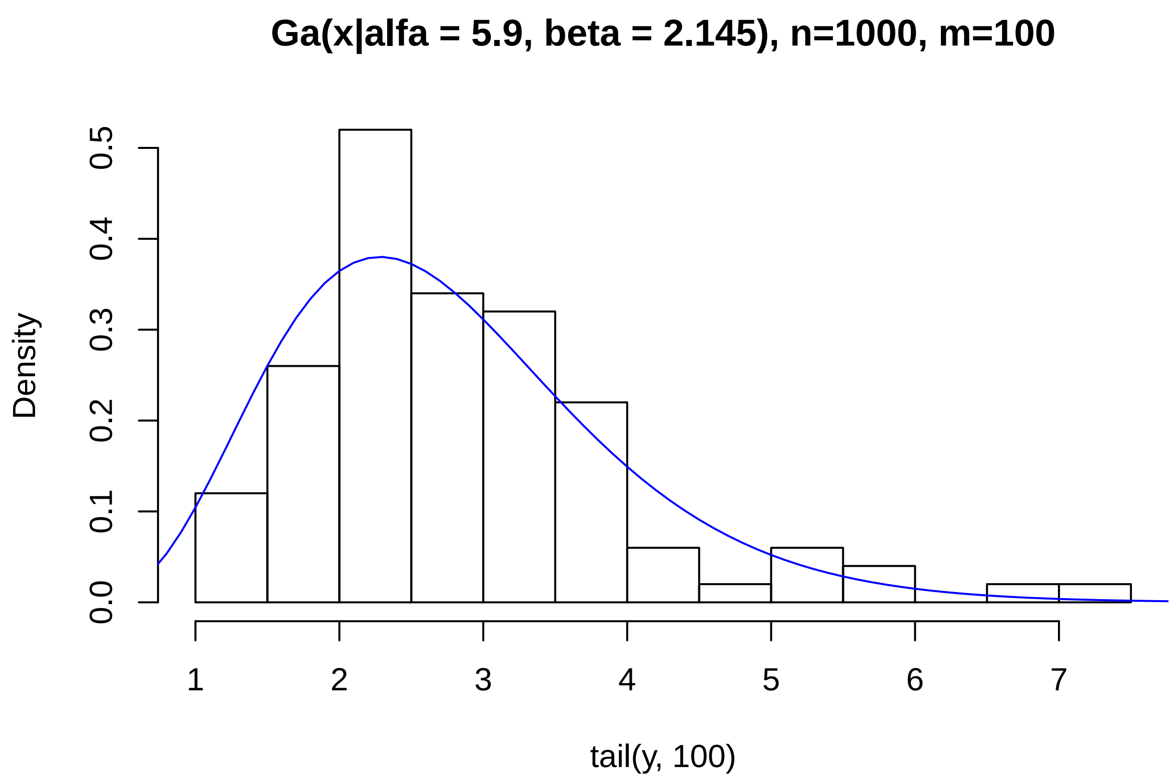
```
## [1] 2.919349
```

Nuevamente los valores empíricos y reales son cercanos, en este caso, la media empírica se alejó más que en el ejercicio anterior con una diferencia de 0.16.

Para $n = 5000$ y $m = 100$ extras tenemos:

```
set.seed(151)
y <- algo(x0 = 1, alfa, beta, n = 5000+100)

hist(tail(y,100),freq = F, main = "Ga(x|alfa = 5.9, beta = 2.145), n=1000, m=100")
  curve(dgamma(x,shape=5.9,rate=2.145),0,10,add = T,col="blue")
```




```
#Cuantil 0.37 real
qgamma(0.37,shape = alfa,rate = beta)
```

```
## [1] 2.250377
```

```
#Cuantil empírico
quantile(tail(y,100),0.37)
```

```
##      37%
## 2.307167
```

```
#Media real
alfa/beta
```

```
## [1] 2.750583
```

```
#Media empírica
mean(tail(y,100))
```

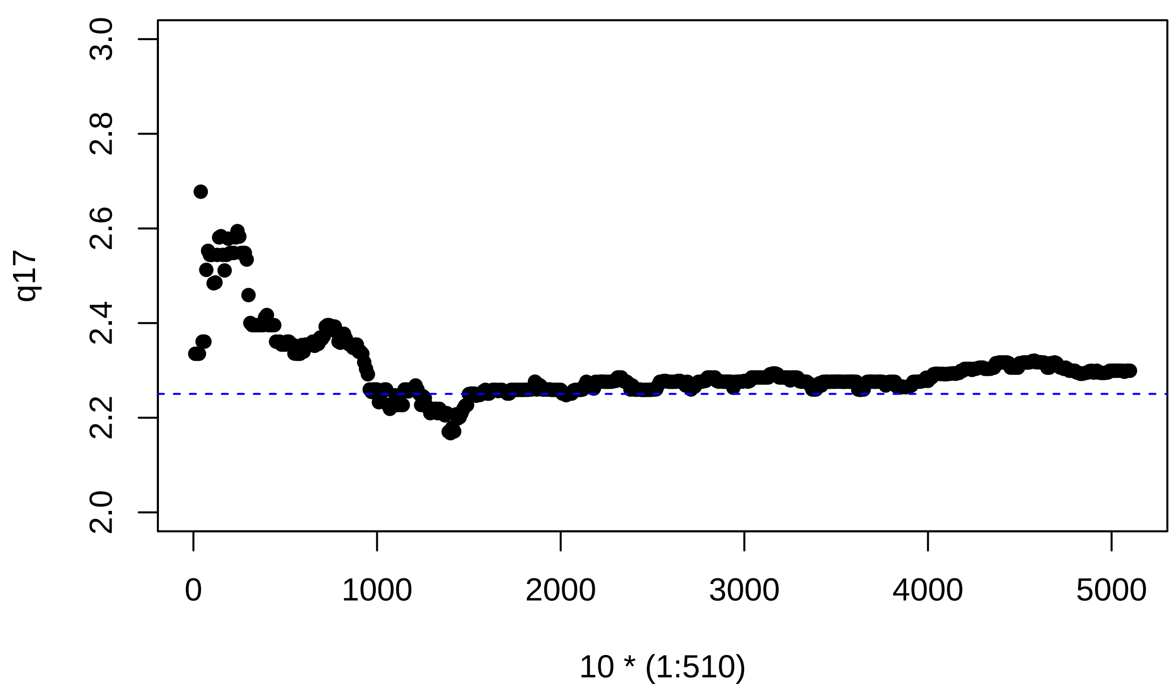
```
## [1] 2.91015
```

Observamos nuevamente que los valores empíricos parecen una buena aproximación a los valores reales.

Para verificar la convergencia del cuantil 0.37 se irá incrementando de 10 en 10 el vector utilizado para calcular el cuantil.

Usando $n + m = 5000 + 100$

```
q37 <- lapply(1:510,FUN = function(l){
  quantile(y[1:(10*l)],0.37)
})
q17 <- unlist(q37)
plot(10*(1:510),q17,pch=16,ylim = c(2,3))
#Línea horizontal con el valor real del cuantil
abline(h = qgamma(0.37,shape = alfa, rate = beta),col = "blue",lty = 2)
```

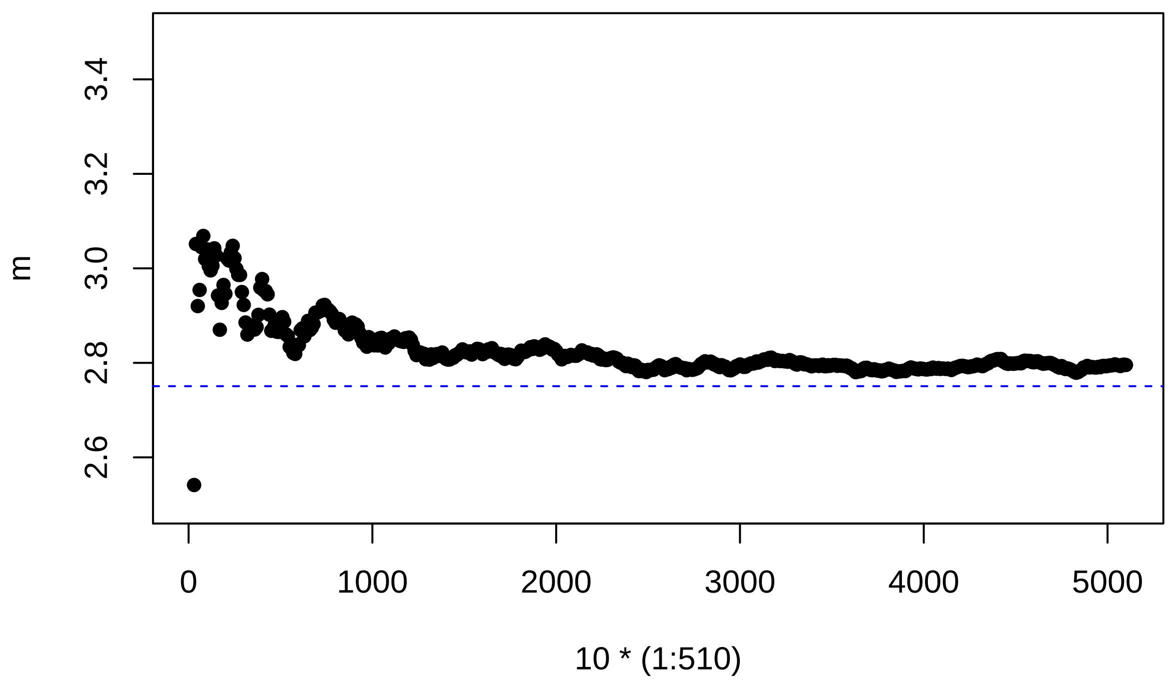


La línea punteada en la gráfica representa el valor verdadero del parámetro. Al principio (a la izquierda) de la gráfica, el cuantil calculado se observa lejos de valor real y a medida que se generan más valores, se observa cierta convergencia, cuando menos el cuantil empírico se va pegando al valor real.

Verifiquemos ahora el caso de la media. De forma análoga se irá incrementado de 10 en 10 el vector utilizado para calcular la media.

Usando $n + m = 5000 + 100$

```
m <- lapply(1:510,FUN = function(l){
  mean(y[1:(10*l)])
})
m <- unlist(m)
plot(10*(1:510),m,pch=16,ylim = c(2.5,3.5))
abline(h = alfa/beta,col = "blue",lty = 2)#línea horizontal con el valor real de la media
```

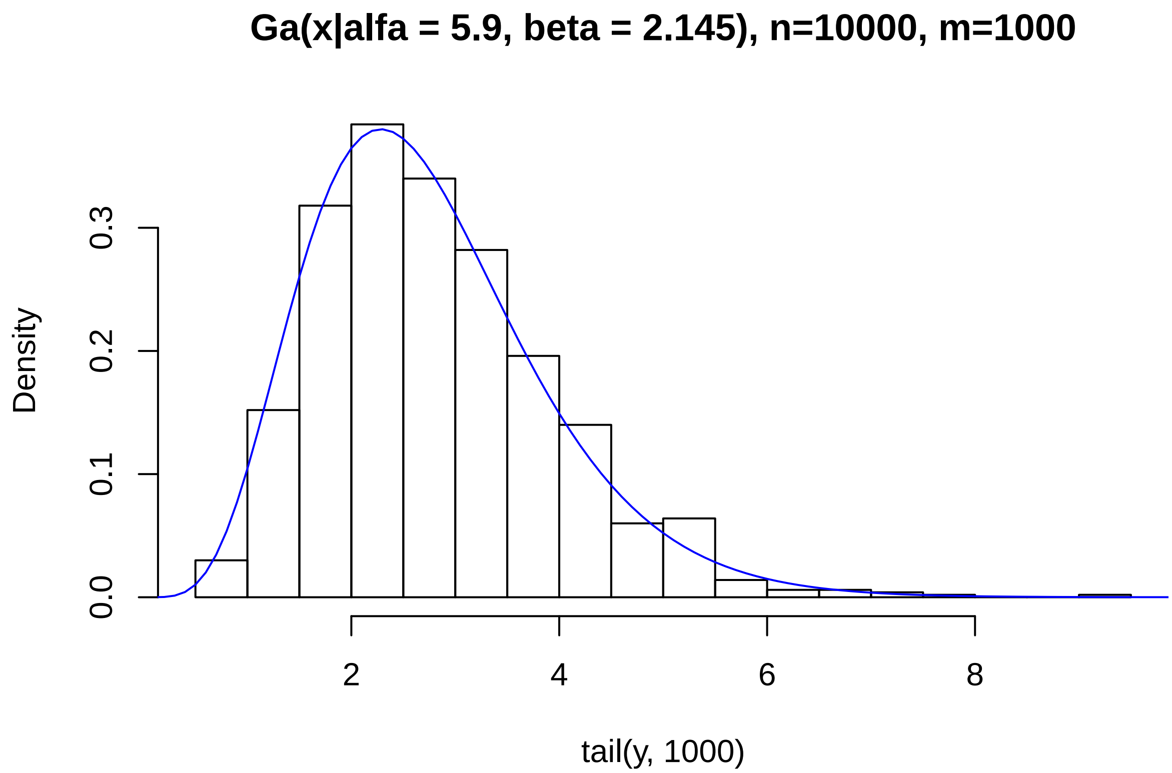


En este caso también se observa como la gráfica se va acercando (está convergiendo) al valor real de la media a medida que se generan más valores. Esto significa que entre más grande sea de elemento que contiene la cadena, los valores generados son más cercanos a los valores de la distribución que se quiere generar.

Sólo por curiosidad, agregaré una gráfica más. Aunque los histogramas mostrados anteriormente presentan más o menos la misma forma que la densidad real, pienso que 100 valores parecen ser pocos para poder observar la densidad empírica, sobre todo porque el número de valores no permite tener muchos grupos en el histograma. Para generar la siguiente gráfica, uso $n = 10000$ y $m = 1000$.

```
set.seed(145)
y <- algo(x0 = 1, alfa, beta, n = 10000+1000)

hist(tail(y,1000),freq = F, breaks = 20, main = "Ga(x|alfa = 5.9, beta = 2.145), n=10000, m=1000")
  curve(dgamma(x,shape=5.9,rate=2.145),0,10,add = T,col="blue")
```




```r
#Cuantil 0.37 real
qgamma(0.37,shape = alfa,rate = beta)
```

```
## [1] 2.250377
```

```r
#Cuantil empírico
quantile(tail(y,1000),0.37)
```

```
##      37%
## 2.300306
```

```r
#Media real
alfa/beta
```

```
## [1] 2.750583
```

```r
#Media empírica
mean(tail(y,1000))
```

```
## [1] 2.823777
```