

Taller 2

Implementación de distribuciones en gamlss

Nombre: _____

Para consultar todo el material de apoyo para este taller visite la página web:

<https://freddy.quarto.pub/dists-gamlss/>

El objetivo de este taller es crear la función familia para una parametrización de la distribución gamma.

1. Existe una parametrización para la distribución gamma donde el parámetro μ coincide con la media de la distribución. La función de densidad para esta distribución es:

$$f(y|\mu, \sigma) = \frac{1}{(\sigma^2\mu)^{1/\sigma^2}} \frac{y^{\frac{1}{\sigma^2}-1} e^{-y/(\sigma^2\mu)}}{\Gamma(1/\sigma^2)}$$

for $y > 0$, where $\mu > 0$ and $\sigma > 0$. Here $E(Y) = \mu$ and $Var(Y) = \sigma^2\mu^2$.

El logaritmo natural de la función $f(y; \mu, \sigma)$ es:

$$\log f(y|\mu, \sigma) = -\frac{1}{\sigma^2} \log(\sigma^2\mu) + \left(\frac{1}{\sigma^2} - 1\right) \log y - \frac{y}{\sigma^2\mu} - \log \Gamma(1/\sigma^2).$$

Calcule las siguientes dos derivadas parciales:

$$dldm = \frac{\partial \log(f(y; \sigma, \mu))}{\partial \mu} =$$

$$dlld = \frac{\partial \log(f(y; \sigma, \mu))}{\partial \sigma} =$$

♦ Al derivar la función $\log(\Gamma(x))$ el resultado es $\psi(x)$, que es la **función digamma**, en R esta función se obtiene con `digamma(x)`.

2. Ahora usted debe crear dos funciones en R para implementar las dos derivadas parciales anteriores. Complete el siguiente código de R con las derivadas anteriores.

```
dldm_manual = function(y, mu, sigma) {  
  dldm <-  
  dldm  
}
```

```
dlld_manual = function(y, mu, sigma) {  
  dlld <-  
  dlld  
}
```

Nota: si quiere revisar si sus funciones son correctas compárelas con las líneas 21 y 22 del enlace:

<https://tinyurl.com/dGAMgamlss>

3. Ahora vamos a poner a prueba la nueva distribución GAM que creamos. Para esto usted debe escribir el siguiente código en R y reemplazar las caritas 😊 por dos valores positivos, los que usted quiera. Puede usar los valores de $\mu=2$ y $\sigma=0.6$ para usar una distribución como la del taller 1.

```
# To load the functions dGAM, pGAM, qGAM, rGAM and GAM.  
source("https://tinyurl.com/58utbsvw")  
source("https://tinyurl.com/4zwpnea4")  
  
# Example 1  
  
# Generating some random values with known mu and sigma  
y <- rGAM(n=1000, mu= 😊 , sigma= 😊 )  
  
# Fitting the model  
require(gamlss)  
mod1 <- gamlss(y~1, sigma.fo=~1, family=GAM)  
  
# Extracting the fitted values for mu and sigma  
# using the inverse link function  
exp(coef(mod1, what="mu"))  
exp(coef(mod1, what="sigma"))
```

¡Felicitaciones, usted acaba de implementar su primera distribución en gamlss!