

Inferencia II - Entrega 2

Daniel Czarniewicz

September 2017

Considere los siguientes datos

Year	Fatal accidents	Passenger deths	Death rate	Miles flown
1976	24	734	0,19	3.863,16
1977	25	516	0,12	4.300,00
1978	31	754	0,15	5.026,67
1979	31	877	0,16	5.481,25
1980	22	814	0,14	5.814,29
1981	21	362	0,06	6.033,33
1982	26	764	0,13	5.876,92
1983	20	809	0,13	6.223,08
1984	16	223	0,03	7.433,33
1985	22	1.066	0,15	7.106,67

El objetivo es modelar la tasa de accidentes fatales por millas voladas. Para esto consideramos el modelo

$$y_i \stackrel{ind}{\sim} \text{Poisson}(x_i \lambda)$$

donde y_i es el número de accidentes fatales en cada año, x_i es el total de millas voladas en cada año y λ es la tasa sobre la que nos interesa realizar inferencia.

Sea el siguiente modelo:

$$\begin{cases} y_i \stackrel{ind}{\sim} \text{Poisson}(x_i \lambda) \\ \lambda \sim \text{Gamma}(a, b) \end{cases} \Rightarrow \begin{cases} p(y_i | x_i \lambda) = \frac{e^{-x_i \lambda} (x_i \lambda)^{y_i}}{y_i!} \mathbf{I}_{[y_i \in \mathbb{N}_0]} \\ p(\lambda) = \frac{b^a}{\Gamma(a)} \lambda^{a-1} e^{-b\lambda} \mathbf{I}_{[\lambda \geq 0]} \end{cases}$$

Utilizando la regla de Bayes obtenemos que:

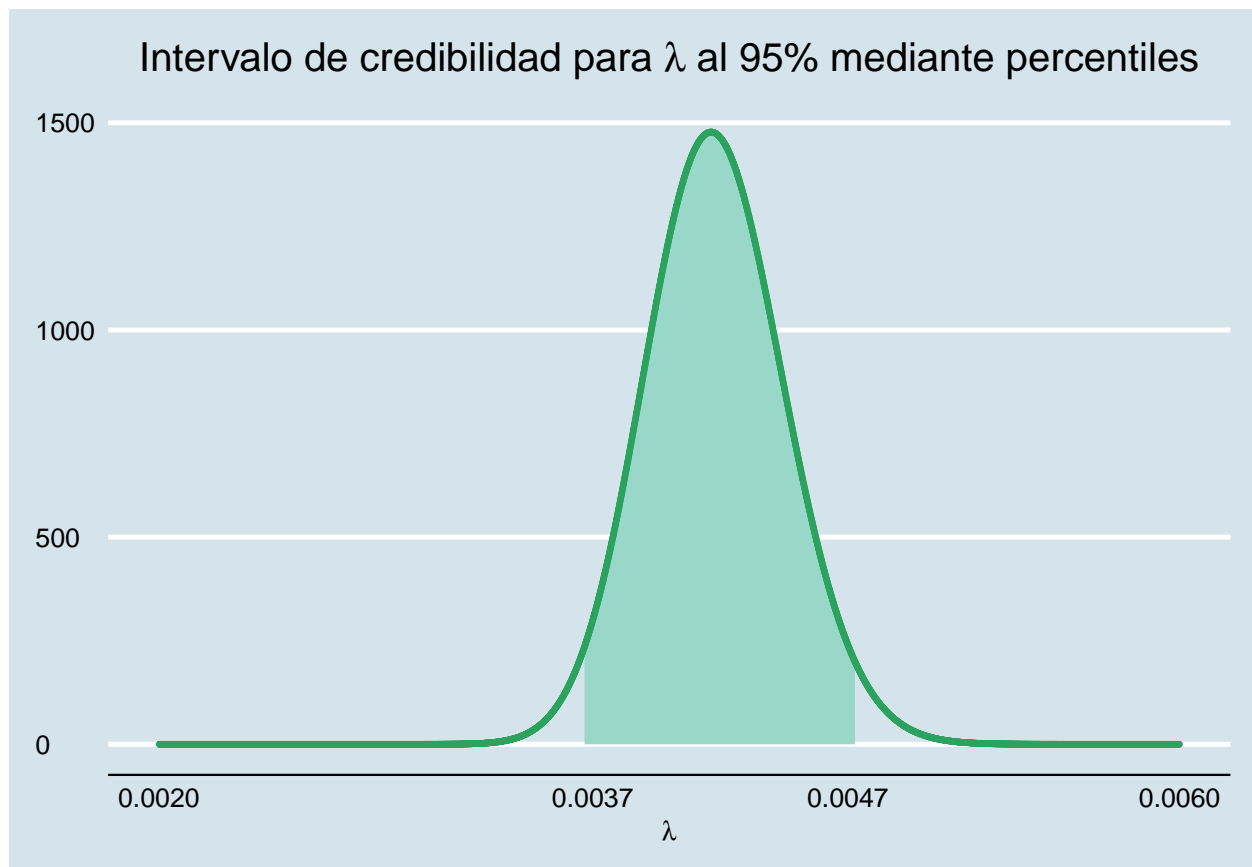
$$\lambda | y_i x_i \sim \text{Gamma} \left(a + \sum_{i=1}^{10} y_i; b + \sum_{i=1}^{10} x_i \right)$$

1. Calcule un intervalo de credibilidad al 95% para λ mediante percentiles.

$$P \left(\text{Gamma}_{(\alpha/2)} < \lambda | y_i x_i < \text{Gamma}_{(1-\alpha/2)} \right) = 1 - \alpha$$

```
a <- b <- 1
sumy <- sum(d$fatal_accidents)
sumx <- sum(d$miles_flown)
alpha <- 0.05
qgamma(c(alpha/2, 1-alpha/2), shape=a+sumy, rate=b+sumx)
```

```
## [1] 0.003667929 0.004727738
```



2. Calcule un intervalo de credibilidad al 95% para λ hallando la región de máxima posterior.

Esta mierda no converge!!

3. Calcule un intervalo de predicción al 95% para $y_{1986}|x_{1986}=8000$ (con cualquier método)

$$\begin{aligned}
 \star p(\tilde{y} | x_i) &= \int_{Rec(\lambda)} p(\tilde{y}; \lambda | x_i) d\lambda = \int_{Rec(\lambda)} p(\tilde{y} | x_i, \lambda) p(\lambda | x_i) d\lambda = \\
 &= \int_{Rec(\lambda)} \frac{e^{-x_i \lambda} (x_i \lambda)^{\tilde{y}}}{\tilde{y}!} \mathbf{I}_{[\tilde{y} \in \mathbb{N}_0]} \frac{b^a}{\Gamma(a)} \lambda^{a-1} e^{-b \lambda} \mathbf{I}_{[\lambda \geq 0]} d\lambda = \\
 &= \frac{b^a x_i^{\tilde{y}}}{\Gamma(a)} \frac{\mathbf{I}_{[\tilde{y} \in \mathbb{N}_0]}}{\tilde{y}!} \int_0^{+\infty} \lambda^{\tilde{y}+a-1} e^{-(b+x_i)\lambda} d\lambda = \frac{b^a x_i^{\tilde{y}}}{\Gamma(a)} \frac{\mathbf{I}_{[\tilde{y} \in \mathbb{N}_0]}}{\tilde{y}!} \frac{\Gamma(\tilde{y}+a)}{(b+x_i)^{\tilde{y}+a}} = \\
 &= \frac{\Gamma(\tilde{y}+a)}{\tilde{y}! \Gamma(a)} \frac{b^a x_i^{\tilde{y}}}{(b+x_i)^{\tilde{y}+a}} \mathbf{I}_{[\tilde{y} \in \mathbb{N}_0]} = \frac{(\tilde{y}+a-1)!}{\tilde{y}! (a-1)!} \left[\frac{b}{b+x_i} \right]^a \left[\frac{x_i}{b+x_i} \right]^{\tilde{y}} \mathbf{I}_{[\tilde{y} \in \mathbb{N}_0]}
 \end{aligned}$$

Por lo tanto:

$$\tilde{y} \sim \text{BN} \left(a; \frac{b}{b+x_i} \right)$$

$$\begin{aligned}
\star p(\tilde{y} | y; x_i) &= \int_{Rec(\lambda)} p(\tilde{y}; \lambda | y; x_i) d\lambda = \int_{Rec(\lambda)} p(\tilde{y} | x_i \lambda; y) p(\lambda | x_i; y) d\lambda = \int_{Rec(\lambda)} p(\tilde{y} | x_i \lambda) p(\lambda | x_i; y) d\lambda = \\
&= \int_{Rec(\lambda)} \frac{e^{-x_i \lambda} (x_i \lambda)^{\tilde{y}}}{\tilde{y}!} I_{[\tilde{y} \in \mathbb{N}_0]} \frac{\left(\sum_{i=1}^{10} x_i + b \right)^{\sum_{i=1}^{10} y_i + a}}{\Gamma \left(\sum_{i=1}^{10} y_i + a \right)} \lambda^{\sum_{i=1}^{10} y_i + a - 1} \exp \left\{ - \left(\sum_{i=1}^{10} x_i + b \right) \lambda \right\} I_{[\lambda \geq 0]} d\lambda = \\
&= \left[\frac{I_{[\tilde{y} \in \mathbb{N}_0]} x_i^{\tilde{y}}}{\tilde{y}!} \right] \left[\frac{\left(\sum_{i=1}^{10} x_i + b \right)^{\sum_{i=1}^{10} y_i + a}}{\Gamma \left(\sum_{i=1}^{10} y_i + a \right)} \right] \underbrace{\int_0^{\infty} \lambda^{\sum_{i=1}^{10} y_i + a + \tilde{y} - 1} \exp \left\{ - \left(\sum_{i=1}^{10} x_i + b + x_i \right) \lambda \right\} d\lambda}_{\text{kernel de una Gamma} \left(\sum_{i=1}^{10} y_i + a + \tilde{y}; \sum_{i=1}^{10} x_i + b + x_i \right)} = \\
&= \left[\frac{I_{[\tilde{y} \in \mathbb{N}_0]} x_i^{\tilde{y}}}{\tilde{y}!} \right] \left[\frac{\left(\sum_{i=1}^{10} x_i + b \right)^{\sum_{i=1}^{10} y_i + a}}{\Gamma \left(\sum_{i=1}^{10} y_i + a \right)} \right] \left[\frac{\Gamma \left(\sum_{i=1}^{10} y_i + a + \tilde{y} \right)}{\left(\sum_{i=1}^{10} x_i + b + x_i \right)^{\sum_{i=1}^{10} y_i + a + \tilde{y}}} \right] =
\end{aligned}$$

Asumiendo que $a \in \mathbb{N}$ y $b \in \mathbb{N}$:

$$\begin{aligned}
&= I_{[\tilde{y} \in \mathbb{N}_0]} \left[\frac{\left(\sum_{i=1}^{10} y_i + a + \tilde{y} - 1 \right)!}{\tilde{y}! \left(\sum_{i=1}^{10} y_i + a - 1 \right)!} \right] \left[\frac{\sum_{i=1}^{10} x_i + b}{\sum_{i=1}^{10} x_i + b + x_i} \right]^{\sum_{i=1}^{10} y_i + a} \left[\frac{x_i}{\sum_{i=1}^{10} x_i + b + x_i} \right]^{\tilde{y}} = \\
&= I_{[\tilde{y} \in \mathbb{N}_0]} \left(\frac{\sum_{i=1}^{10} y_i + a + \tilde{y} - 1}{\sum_{i=1}^{10} y_i + a - 1} \right) \left[\frac{\sum_{i=1}^{10} x_i + b}{\sum_{i=1}^{10} x_i + b + x_i} \right]^{\sum_{i=1}^{10} y_i + a} \left[\frac{x_i}{\sum_{i=1}^{10} x_i + b + x_i} \right]^{\tilde{y}}
\end{aligned}$$

Por lo tanto:

$$\tilde{y} | y \sim \text{BN} \left(\sum_{i=1}^{10} y_i + a; \frac{\sum_{i=1}^{10} x_i + b}{\sum_{i=1}^{10} x_i + b + x_i} \right)$$

```
xi <- 8000
qnbino(c(alpha/2, 1-alpha/2), size=sumy+a, prob=((b+sumx)/(b+sumx+xi)))
```

```
## [1] 22 46
```

Considere el siguiente código de R. Los objetos `a1` y `b1` representan los parámetros en la posterior $p(\lambda|y)$

```
N <- 1e3
a1 <- "???"
b1 <- "???"

slope_fn <- Vectorize(
  function(lam, xs) {
    yrep <- rpois(length(xs), lambda=lam*xs)
    m <- lm(log(yrep) ~ xs)
    coef(m)[2]
  },
  vectorize.args = 'lam')

slp.sims <- data_frame( lambda=rgamma(N, a1, b1) ) %>%
  mutate(slp = slope_fn(lam=lambda, xs = d$miles_flow))
```

4. Describa el código anterior. ¿Qué hace la función `slope_fn`? (la parte de `Vectorize` no importa).

`slope_fn` toma dos vectores numéricos, `lam` y `xs`. Genera el vector `yrep` con `length(xs)` simulaciones de una distribución Poisson con esperanza y varianza igual a `lam*xs`. Luego estima un modelo lineal con variable dependiente `log(yrep)` contra una constante y `xs`. Devuelve el valor del parámetro asociado al regresor `xs`. Luego `Vectorize` vectoriza la función sobre el conjunto de lambdas especificado por el usuario.

5. Complete el código con los valores para `a1`, `b1` y ejecútalo. ¿Qué queda en el objeto `slp.sims`?

```
N <- 1e3
a <- b <- 1
a1 <- a+sumy
b1 <- b+sumx

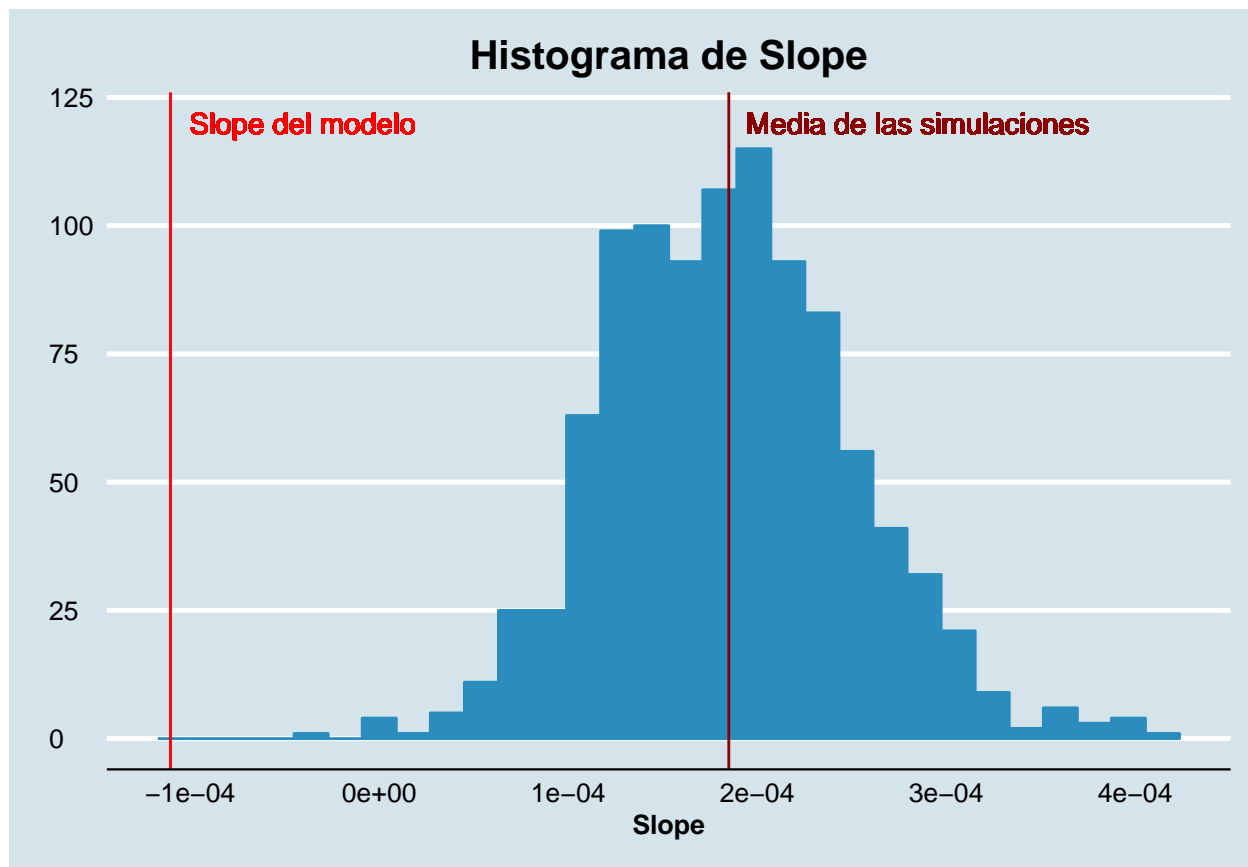
set.seed(1234)
slope_fn <- Vectorize(
  function(lam, xs) {
    yrep <- rpois(length(xs), lambda = lam*xs)
    m <- lm(log(yrep) ~ xs)
    coef(m)[2]
  },
  vectorize.args = 'lam')

slp.sims <- data_frame(lambda=rgamma(N, shape=a1, rate=b1)) %>%
  mutate(slp=slope_fn(lam=lambda, xs=d$miles_flow))
```

Devuelve un `data.frame` con 1000 filas y 2 columnas. La primera columna corresponde a los valores de `lambda` simulados con `rgamma(N, shape=a1, rate=b1)`. La segunda columna contiene los valores de las pendientes calculadas mediante la función `slope_fn`.

Lambda	Slope
0,003853	0,000198
0,004258	0,000255
0,004270	0,000172
0,003977	0,000322
0,004310	0,000137
0,004019	0,000208
0,004442	0,000236
0,004006	0,000194

6. Realiza un histograma de la variable `slp` y agrega una línea vertical de color rojo con la pendiente del modelo `lm(log(fatal_accidents) ~ miles_flown, data=d)`



7. ¿Qué nos informa el dibujo sobre el modelo que estamos estimando?

Que es un culo!