Examen final - Otoño 2020

Alfredo Garbuno Iñigo

Entrega: Enviar la carpeta que el codigo de solución (.Rmd y funciones auxiliares) a mas tardar el 15 de Diciembre antes de las 12:00pm (mediodia), por correo electrónico con el título fundamentos-final, un solo documento por equipo. No se aceptarán entregas extemporáneas. Será mejor entregar un examen resuelto parcialmente, que no entregar nada.

Instrucciones:

- Tus respuestas deben ser claras y debes explicar los resultados, incluye también tus procedimientos/código de manera ordenada, y el código comentado.
- Se evaluará la presentación de resultados (calidad de las gráficas, tablas,...), revisa la sección de visualización en las notas.
- Las sesiones del Martes 8 y Jueves 10 de Diciembre a las 10 am, serán espacios para resolver dudas que puedan surgir del exámen.
- No pueden compartir soluciones entre diferentes equipos, o alumnos del grupo 001 de esta misma materia.
- Al entregar este examen afirmas que el trabajo se realizó sólo con tu compañero de equipo. El material que utilizaste para apoyarte consistió de las notas en clase (pdf en canvas), el codigo fuente de las notas en el repositorio de Github.
- Al entregar estás dando tu consentimiento para que bajo sospecha y suficiente evidencia de copia se anule tu evaluación.

Preparación de ambiente

Asegurate de tener instalado los paquetes que usamos más en las notas del curso. En particular, si usas `renv` como manejador de ambientes puedes instalarlos con las instrucciones de abajo. Sólo necesitarías descomentarlas.

```
# renv::install("tidyverse")
# renv::install("patchwork")
# renv::install("nullabor")
# renv::install("scales")
# renv::install('diegovalle/mxmaps')
# renv::install("nleqslv")
# renv::snapshot()

# Escribe las claves unicas de ambos miembros del equipo, para generar una
# semilla de numeros aleatorios.
claves_unicas <- c(151280, 150370)
set.seed(min(claves_unicas))
```

Modelos de conteo

En el curso hemos estudiado las variables aleatorias Gaussianas para modelar eventos aleatorios compuestos de pequeños, pero controlados, efectos. También hemos utilizado variables aleatorias Binomiales para modelar tasas de éxito de algún evento binario de interés. En el contexto Bayesiano, hemos utilizado las distribuciones Beta, Gamma-Inversa, y Normal para realizar análisis conjugado con estos modelos.

En este mini-proyecto, ilustraremos otra familia de distribuciones muy cómunes en la práctica. En particular, veremos la distribución **Poisson** como unmodelo de conteo. Es decir, una variable aleatoria Poisson nos sirve para modelar el número de ocurrencias de un evento en un periodo (tiempo) o área (espacio) base.

Decimos que $x|\theta \sim \mathsf{Poisson}(\theta)$ si los eventos ocurren de manera independiente y a una tasa constante. La función de masa de probabilidad esta dada por

$$p(X = k \,|\, \theta) = \frac{\theta^k \, e^{-\theta}}{k!},$$

donde sabemos que

$$\mathbb{E}[x|\theta] = \theta, \qquad \mathbb{V}[x|\theta] = \theta$$

Al examinar la base de la función de masa de probabilidad notamos que un candidato para un anâlisis conjugado es una distribución Gamma. Es decir, un candidato *natural* para una distribución previa para θ es

$$\theta \sim \mathsf{Gamma}(\alpha, \beta),$$

donde la densidad está dada por

$$p(\theta) \propto \theta^{\alpha-1} \, e^{-\beta \, \theta},$$

y tenemos los siguientes momentos

$$\mathbb{E}[\theta] = \frac{\alpha}{\beta}, \qquad \mathbb{V}[\theta] = \frac{\alpha}{\beta^2}.$$

Pregunta 1) Para una muestra $X_1, \ldots, X_n \overset{iid}{\sim} \mathsf{Poisson}(\theta)$, determina la distribución posterior de θ, y calcula media y varianza de la distribución posterior. ¿Podríamos escribir la media posterior como un promedio ponderado entre datos e información previa? ¿Cómo interpretas los hiper-parámetros (α, β)?

$$\begin{aligned}
\Pi(\theta \mid x_1, \ldots, x_n) &\propto \Pi(x_1, \ldots, x_n \mid \theta) \cdot \Pi(\theta) \\
&\propto \prod_i \Pi(x_i \mid \theta) \cdot \Pi(\theta) \\
&\propto \prod_i \frac{\theta^{x_i} e^{-\theta}}{x_i!} \cdot \theta^{\alpha-1} e^{-\beta\theta} \\
&\propto \frac{\theta^{\sum x_i} e^{-n\theta}}{\prod x_i!} \cdot \theta^{\alpha-1} e^{-\beta\theta} \\
&\propto \theta^{\alpha+\sum x_i - 1} e^{-\theta(\beta+n)}
\end{aligned}$$

De lo anterior se puede ver que

$$\Pi(\theta \mid x_1, \ldots, x_n) \sim Gamma(\alpha + \sum x_i, \beta + n)$$

Así,

$$E[\theta \mid x_1, \ldots, x_n] = \frac{\alpha + \sum x_i}{\beta + n} Var[\theta \mid x_1, \ldots, x_n] = \frac{\alpha + \sum x_i}{(\beta + n)^2}$$

Sí se puede ver como un promedio entre los datos y el parámetro α. Entre más datos sean más peso tendrán en la posterior. En cambio si los datos $n \to 0$, la posterior será igual a la distribución a priori.

Otra variable aleatoria de conteo relevante es la **Binomial Negativa.** Esta distribución sirve para modelar el número de éxitos en una secuencia de experimentos Bernoulli antes de encontrar un número específico de fracasos.

Decimos que $X|\alpha, \beta \sim \mathsf{Neg\text{-}Bin}(\alpha, \beta)$, donde X es el número de éxitos que contamos antes de α fracasos, cuando cada fracaso ocurre con probabilidad $\frac{\beta}{\beta+1}$. La función de masa de probabilidad se escribe

$$p(X = k \,|\, \alpha, \beta) = \binom{\alpha + k - 1}{k} \left(\frac{\beta}{\beta + 1}\right)^{\alpha} \left(\frac{1}{\beta + 1}\right)^{k}.$$

Nota que

$$\binom{\alpha + k - 1}{k} = \binom{\alpha + k - 1}{\alpha - 1},$$

es decir, el número de formas que puedes acomodar $\alpha - 1$ fracasos es igual al número de formas que puedes acomodar k éxitos cuando realizaste $\alpha + k - 1$ experimentos y todos los experimentos son independientes. Por otro lado, la definición es

$$\binom{\alpha + k - 1}{k} = \frac{(\alpha + k - 1)!}{k!\,(\alpha - 1)!}.$$

donde $k! = k \times k - 1 \times k - 2 \times \cdots \times 1$, y la función Gamma satisface

$$\Gamma(\alpha) = (\alpha - 1)!.$$

Pregunta 2) Bajo el modelo conjugado que escribiste en la pregunta 1, calcula la **distribución predictiva previa** para una observación Poisson. Es decir, calcula

$$p(y) = \int \mathsf{Poisson}(y \,|\, \theta)\mathsf{Gamma}(\theta \,|\, \alpha, \beta)\, \mathrm{d}\theta.$$

$$\begin{aligned} p(y) &= \int_0^\infty \frac{\theta^y e^{-\theta}}{y!} \cdot \frac{\beta^\alpha \theta^{\alpha-1} e^{-\beta\theta}}{\Gamma(\alpha)} d\theta \\ &= \frac{\beta^\alpha}{\Gamma(\alpha) y!} \int_0^\infty \theta^{y+\alpha-1} e^{-\theta(\beta+1)} d\theta \end{aligned}$$

Sabemos que:

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$$

Entonces si:

$$t = \theta(\beta + 1) \Rightarrow \theta = \frac{t}{\beta + 1} d\theta = \frac{1}{\beta + 1} dt$$

Así:

$$
\begin{aligned}
&= \frac{\beta^\alpha}{\Gamma(\alpha)y!}\int_0^\infty \frac{t}{\beta+1}^{y+\alpha-1} e^{-t}\frac{1}{\beta+1}dt \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)y!}\cdot\frac{1}{(\beta+1)^{y+\alpha}}\int_0^\infty t^{y+\alpha-1}e^{-t}dt \\
&= \frac{\beta^\alpha}{\Gamma(\alpha)y!}\cdot\frac{1}{(\beta+1)^{y+\alpha}}\Gamma(y+\alpha) \\
&= \frac{\Gamma(y+\alpha)}{\Gamma(\alpha)y!}\frac{\beta^\alpha}{(\beta+1)^{y+\alpha}} \\
&= \frac{(y+\alpha-1)!}{(\alpha-1)!y!}\left(\frac{\beta}{\beta+1}\right)^\alpha\left(\frac{1}{\beta+1}\right)^y \\
&= \binom{y+\alpha-1}{y}\left(\frac{\beta}{\beta+1}\right)^\alpha\left(\frac{1}{\beta+1}\right)^y
\end{aligned}
$$

Verifica tu cálculo utilizando las reglas probabilidad condicional. En especifico, utiliza

$$p(y) = \frac{p(y|\theta)p(\theta)}{p(\theta|y)}.$$

Tenemos que:

$$
\begin{aligned}
p(y) &= \frac{\frac{\theta^y e^{-\theta}}{y!}\frac{\beta^\alpha\theta^{\alpha-1}e^{-\beta\theta}}{\Gamma(\alpha)}}{\frac{(\beta+1)^{\alpha+y}\theta^{\alpha+y-1}e^{-(\beta+1)\theta}}{\Gamma(\alpha+y)}} \\
&= \frac{\Gamma(\alpha+y)}{\Gamma(\alpha)y!}\frac{\beta^\alpha}{(\beta+1)^{\alpha+y}}\theta^0 e^0 \\
&= \frac{(y+\alpha-1)!}{(\alpha-1)!y!}\left(\frac{\beta}{\beta+1}\right)^\alpha\left(\frac{1}{\beta+1}\right)^y \\
&= \binom{y+\alpha-1}{y}\left(\frac{\beta}{\beta+1}\right)^\alpha\left(\frac{1}{\beta+1}\right)^y
\end{aligned}
$$

¿Qué distribución marginal tiene y bajo el modelo conjugado?

Es una distrubución binomial negativa, por lo tanto

$$p(y) \sim BinNeg(\alpha, \beta)$$

En la práctica, es útil extender el modelo Poisson como sigue

$$x_i|t_i, \theta \sim \mathsf{Poisson}(\lambda_i), \tag{1}$$
$$\lambda_i = t_i\theta, \tag{2}$$

donde la tasa de ocurrencia λ_i ha sido descompuesta en un producto que incorpora la exposición t_i y una tasa de ocurrencia por unidades expuestas θ. En este contexto usualmente tenemos observaciones para x_i y t_i pues conocemos el parámetro de exposición. Por ejemplo, si x_i es el número de personas que se enferman de gripe en la i-ésima ciudad en un año, entonces θ denota la tasa anual por persona de enfermarse de gripe en una población de tamaño t_i.

Pregunta 3) Supongamos que tenemos datos $X_1, \ldots, X_n \sim \mathsf{Poisson}(\lambda_i)$, con $\lambda_i = t_i\theta$ para $i = 1, \ldots, n$. Utilizando el modelo conjugado, ¿cuál es la distribución posterior de θ?

$$
\begin{aligned}
\Pi(\theta \mid x_1, \ldots, x_n) &\propto \Pi(x_1, \ldots, x_n \mid \theta) \cdot \Pi(\theta) \\
&\propto \prod_i \Pi(x_i \mid \theta) \cdot \Pi(\theta) \\
&\propto \prod_i \frac{(\theta t_i)^{x_i} e^{-\theta t_i}}{x_i!} \cdot \theta^{\alpha-1} e^{-\beta\theta} \\
&\propto \theta^{\sum x_i} e^{-\theta \sum t_i} \cdot \theta^{\alpha-1} e^{-\beta\theta} \\
&\propto \theta^{\alpha+\sum x_i - 1} e^{-\theta(\beta+\sum t_i)}
\end{aligned}
$$

Por lo tanto, la distribución posterior es:

$$
\Pi(\theta \mid x_1, \ldots, x_n) \sim Gamma(\alpha + \sum x_i, \beta + \sum t_i)
$$

Caso de estudio: Tasas de mortalidad

El INEGI publica para cada año los registros de fallecimiento junto con la causa principal de muerte. En esta sección utilizaremos los modelos descritos anteriormente para inferir tasa de fallecimiento por Neumonía para cada uno de los municipios/delegaciones del país. Contamos con los últimos 5 años de los registros de defunción.

Carga y preparación de datos

Pregunta 4) Empecemos explorando los datos. Carga los datos para un año que elijas. Encontrarás en los archivos en `datos/poblacion/defunciones/<año>` los registros de defunciones por Neumonía para el `<año>` que escojas.

```
defunciones <- read_csv("./datos_examen/poblacion/defunciones/2019/defunciones_registradas.csv")


str(defunciones)
```

```
## tibble [30,327 x 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ entidad    : chr [1:30327] "01" "01" "01" "01" ...
##  $ municipio  : chr [1:30327] "001" "001" "001" "003" ...
##  $ sexo       : chr [1:30327] "mujer" "hombre" "mujer" "mujer" ...
##  $ edad       : num [1:30327] 55 81 82 83 82 55 0 66 76 75 ...
##  $ edad_grupos: chr [1:30327] "[25,64)" "[64,Inf]" "[64,Inf]" "[64,Inf]" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   entidad = col_character(),
##   ..   municipio = col_character(),
##   ..   sexo = col_character(),
##   ..   edad = col_double(),
##   ..   edad_grupos = col_character()
##   .. )
```

```
map_int(defunciones[,1:5], ~length(unique(.x)) )
```

```
##     entidad   municipio        sexo        edad edad_grupos
##          33         314           3         114           7
```

- El dataset contiene 5 variables y tenemos un total de $30,327$ observaciones.
- Observemos que tenemos 33 entidades, lo cual hace un poco de ruido, ya que, la República Mexicana está constituída por 32 entidades federativas.
- En el dataset se tiene información de 314 municipios, de acuerdo a informaicón declarada por el INEGI hay un total de $2,465$ municipios, por lo que nuestra muestra contiene aproximadamente el 12% de ellos.
- Para el sexo aparecen 3 categorías (intuitivamente diríamos que es Masculino, Femenino y No declarado).
- Tenemos 7 grupos de edad, los cuales son los siguientes:

```
defunciones$edad_grupos %>% unique
```

```
## [1] "[25,64)"  "[64,Inf]" "[0,3)"    "[3,6)"    "[6,12)"   "[18,25)"  "[12,18)"
```

- $[3,6)$
- $[6,12)$
- $[12,18)$
- $[18,25)$
- $[25,64)$
- $[64,\text{inf})$

Para el tema de los 33 estados analicemos los valores:

```
defunciones$entidad %>% unique() %>% sort()
```

```
##  [1] "01" "02" "03" "04" "05" "06" "07" "08" "09" "10" "11" "12" "13" "14" "15"
## [16] "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29" "30"
## [31] "31" "32" "99"
```

Nótese que se tiene una entidad 99, que por intuición indicaríamos que es una Entidad no identificada.

Y ahora para la variable sexo:

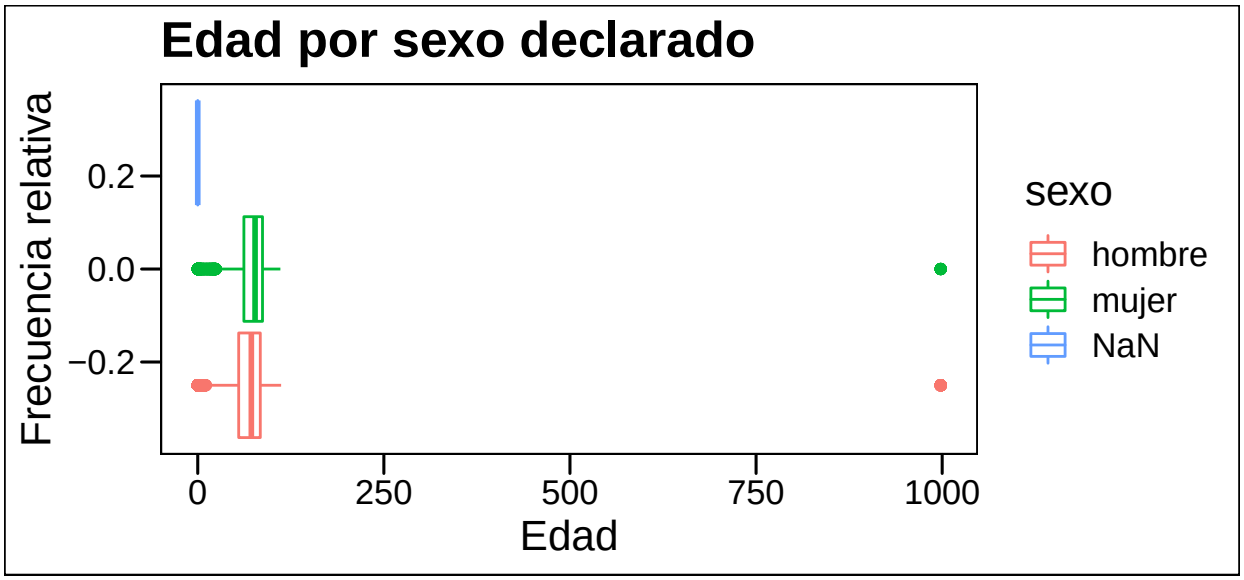
```
defunciones$sexo %>% unique() %>% sort()
```

```
## [1] "hombre" "mujer"  "NaN"
```

Nuestra hipótesis era casi correcta, tenemos un tercer sexo el cual es NaN, que indica nos sugiere que el usuario no se identificó con alguno de los 2 sexos o bien, no quiso declarar esa información.

Ahora analicemos un poco la variabilidad de los datos.

```
defunciones %>%
  ggplot(data = ., aes(x = edad)) +
  geom_boxplot(aes(color = sexo)) +
  ggtitle('Edad por sexo declarado') +
  xlab('Edad') +
  ylab('Frecuencia relativa') +
  ggthemes::theme_base()
```

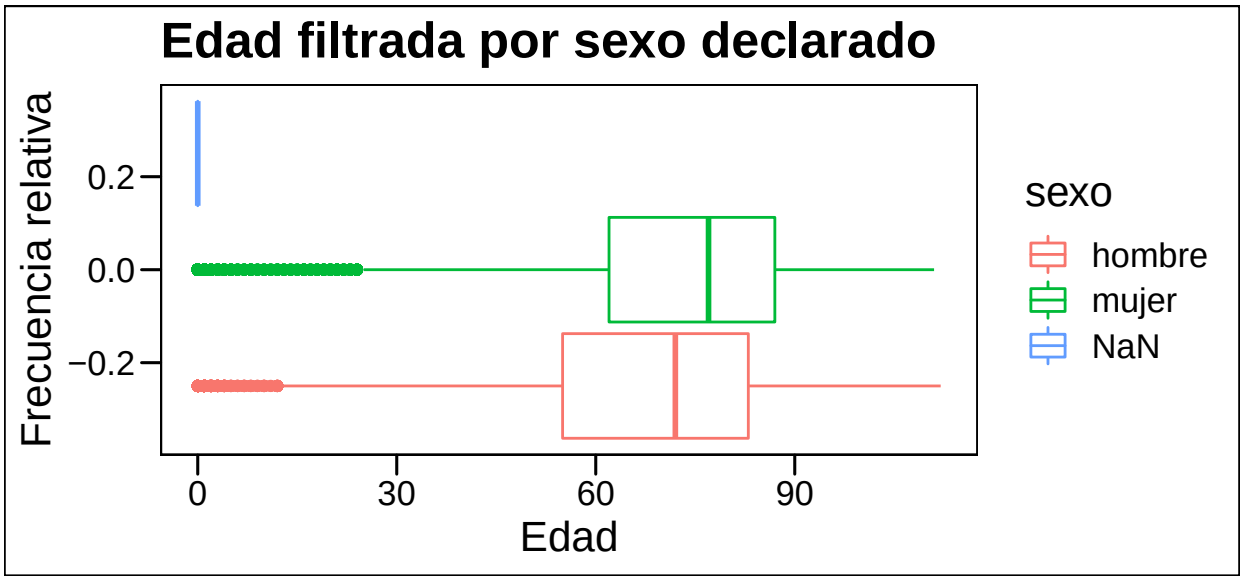



```
defunciones %>% colnames()
```

```
## [1] "entidad"     "municipio"   "sexo"        "edad"        "edad_grupos"
```

Es difícil analizar la dispersión de las edades por sexo debido a la existencia de 2 valores atípicos, los cuales equivalen a la edad 1000 años, lo cual no hace sentido, pues no existe ningún registro de una persona en la historia de la humanidad que haya sobrevivido a 1,000 años. Suponemos que para dichas observaciones el individuo no quiso declarar su edad o bien hubo un error en el registro de la captura del dato. A continuación analizaremos las edades sin dichos valores.

```
defunciones %>%
  filter(edad < 250) %>%
  ggplot(data = ., aes(x = edad)) +
  geom_boxplot(aes(color = sexo)) +
  ggtitle('Edad filtrada por sexo declarado') +
  xlab('Edad') +
  ylab('Frecuencia relativa') +
  ggthemes::theme_base()
```



Realizando el filtro correspondiente es un poco más "sencillo" (es una palabra que no nos gusta utilizar, ya que es relativo a la persona) analizar la dispersión de la edad para las personas: * La mediana de edad para mujeres es mayor que para los hombres, sin embargo, ambas distribuciones son muy similares. * Para el sexo no declarado la edad únicamente es 0, por lo que, para dichas observaciones el individuo no quiso declarar su sexo ni edad.

En relación a la variable `entidad`, estos son nuestros comentarios:

```
defunciones %>%
  select(entidad) %>%
  table() %>%
  sort(decreasing = TRUE)
```

```
## .
##   09   14   15   19   30   21   11   08   07   26   16   02   05   28   25   31
## 4062 3427 3179 2062 1582 1378 1255 1173 1073  932  856  823  699  683  677  659
##   20   24   17   22   12   13   10   27   32   01   18   23   06   29   99   04
##  653  618  492  480  414  404  385  359  359  311  263  231  213  196  158  137
##   03
##  134
```

Es difícil saber qué Estado representa cada número, ya que, no se cuenta con un diccionario de datos. Por ahora solamente se puede indicar que el estado 9, es el que tiene un mayor peso en el dataset.

Pregunta 5) De igual forma, carga los datos de población que encontrarás en `datos_examen/poblacion/demograficos`. Por el momento, no necesitamos los grupos de edad (aunque despúes los utilizaremos). Por ahora escribe el codigo necesario para calcular el tamaño de la población en cada uno de los municipios.

```
poblacion <- read_csv("./datos_examen/poblacion/demograficos/poblacion_municipios_edad.csv")

poblacion <- poblacion %>%
  mutate(pob_tot = p_0a2+p_3a5+p_6a11+p_12a17+p_18a24+p_25a64+pob65_mas)
```

Pregunta 6) Ahora necesitamos *cruzar* las tablas de defunciones y población para crear una tabla con ambos registros. Para esto necesitarás la función `dplyr::full_join`.

```
# Primero se agrupan los datos de defunciones por municipio
defunciones_mun <- defunciones %>%
  group_by(entidad, municipio) %>%
  count(name = 'defunciones')

datos = poblacion %>%
  left_join(defunciones_mun, by=c('entidad', 'municipio'))
```

Pregunta 7) Con esto tendrás conocimiento de cómo cargar la información relevante (número de defunciones y población total en cada municipio). Sin embargo, tenemos información para las defunciones de los últimos 5 años. Carga la información que encontrarás en `/defunciones/` y agrupa de tal forma que tengas una tabla como la anterior. **Importante:** Para fines de este proyecto no necesitamos los conteos por año, sólo el agrupado. Es decir, el número de defunciones totales de los 5 años por municipio.

```
# En la funcion cargar_defunciones, en caso de error
# Por favor escribir como parámetro la dirección donde se entuentran
# los archivos de defunciones (ej: "./datos_examen/poblacion/defunciones/")
cargar_defunciones <- function(path="./datos_examen/poblacion/defunciones/"){
  x <- list.files(path, full.names=TRUE, recursive = TRUE)
  def_acum <- NULL
  for (file in x){
    def <- read_csv(file)
```

```
    def_acum <- def_acum %>% bind_rows(def)
  }
  return(def_acum)
}
defunciones <- cargar_defunciones()

defunciones_mun <- defunciones %>%
  group_by(entidad, municipio) %>%
  count(name = 'muertes_neum')

full_data = poblacion %>%
  left_join(defunciones_mun, by=c('entidad', 'municipio')) %>%
  replace_na(list(muertes_neum=0)) %>%
  mutate(tasa_mort = muertes_neum/pob_tot)
```

Cálculo de estadístico de interés

Lo que nos interesa en particular son las tasas de mortalidad anual en los municipios del país. Para esto utilizaremos el modelo Poisson que vimos en la primera parte. Si denotamos por y_i el número total de defunciones por neumonia en el i-ésimo municipio; θ_i, la tasa de mortalidad por individuo por año, entonces

$$y_i \,|\, n_i, \theta_i \sim \text{Poisson}(\lambda_i),$$

donde n_i denota la población total del municipio i-ésimo y λ_i la tasa con la que ocurren las muertes por neumonía en el periodo observado para la población del municipio i-ésimo.

Pregunta 8) ¿Cómo escribirías λ_i en función de θ_i?

Ahora, utilizaremos un mapa para ver si podemos observar algún patrón en las tasas de mortalidad por individuo por año θ_i. Por ejemplo, podríamos esperar que algunas zonas del país concentren las tasas mas altas. Por ejemplo, podemos crear mapas con los municipios con las tasas mas bajas y altas. Digamos que sólo queremos ver el 25% mas bajo y alto. Los mapas los obtenemos con las funciones `mxmaps::mxmunicipio_choropleth`.

La estructura que necesita esta función es una tabla con una columna que se llame `region` donde venga el codigo identificador del municipio. Por ejemplo, para el municipio `001` en el estado `24` el codigo de region será `24001`. Otra columna necesaria es el valor con el que "coloreará" el municipio en el mapa y se tiene que llamar `value` y puede ser una variable `Boolean` o `double`.

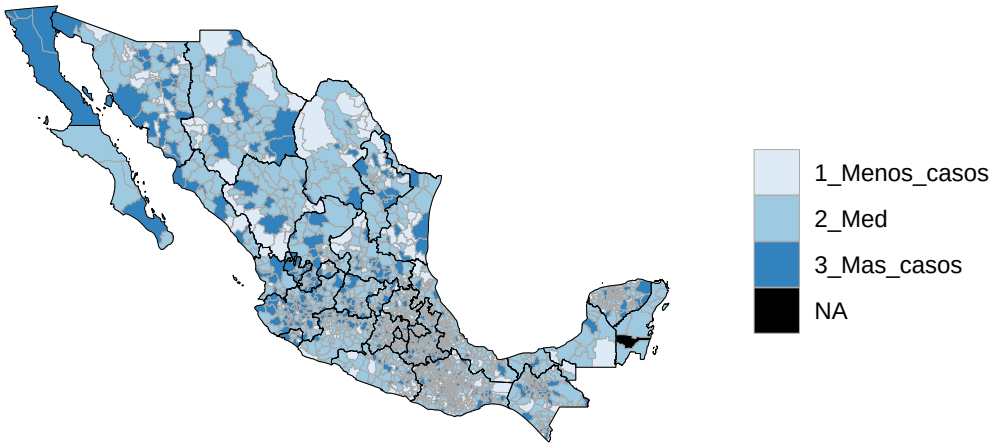
Pista. Para este punto, podrías necesitar la función `dplyr::row_number`. De igual forma podrías ocupar una indicadora para decir cuáles son los municipios con las tasas mas altas y cuáles son los que tienen las mas bajas. Al final, podrías presentar esto como dos mapas separados.

Respuesta: La manera de representar λ_i en función de θ_i es la siguiente:

$$\lambda_i(\theta_i) = 5 * \theta_i * n_i$$

```
datos2 <- full_data %>%
  mutate(region = paste0(entidad,municipio),
         mortalidad = ((muertes_neum)/pob_tot),
         orden = row_number(mortalidad)/nrow(full_data),
         value = ifelse(orden<0.75,
         ifelse(orden<0.25,"1_Menos_casos","2_Med"),
         "3_Mas_casos"))
```

```
mxmunicipio_choropleth(datos2)
```



¿Qué observas? No hay patrón tan claro. Especialmente si observamos lo que sucede en Chihuahua, Durango y Coahuila, donde tenemos municipios de ambas categorías. ¿Cómo puede ser que un mismo estado tenga las tasas mas altas y bajas al mismo tiempo?

¡El problema es el tamaño de muestra! Considera un municipio de 1,000 habitantes. Muy probablemente en 5 años no veamos una muerte por neumonía, lo cual convertiría la tasa observada en 0. Sin embargo, si ocurriera una muerte entonces la tasa sería de 1/5,000 por año, lo cual sería muy elevado con respecto a otros municipios con poblaciones grandes y mayor número de casos.

Inferencia Bayesiana para tasa de mortalidad

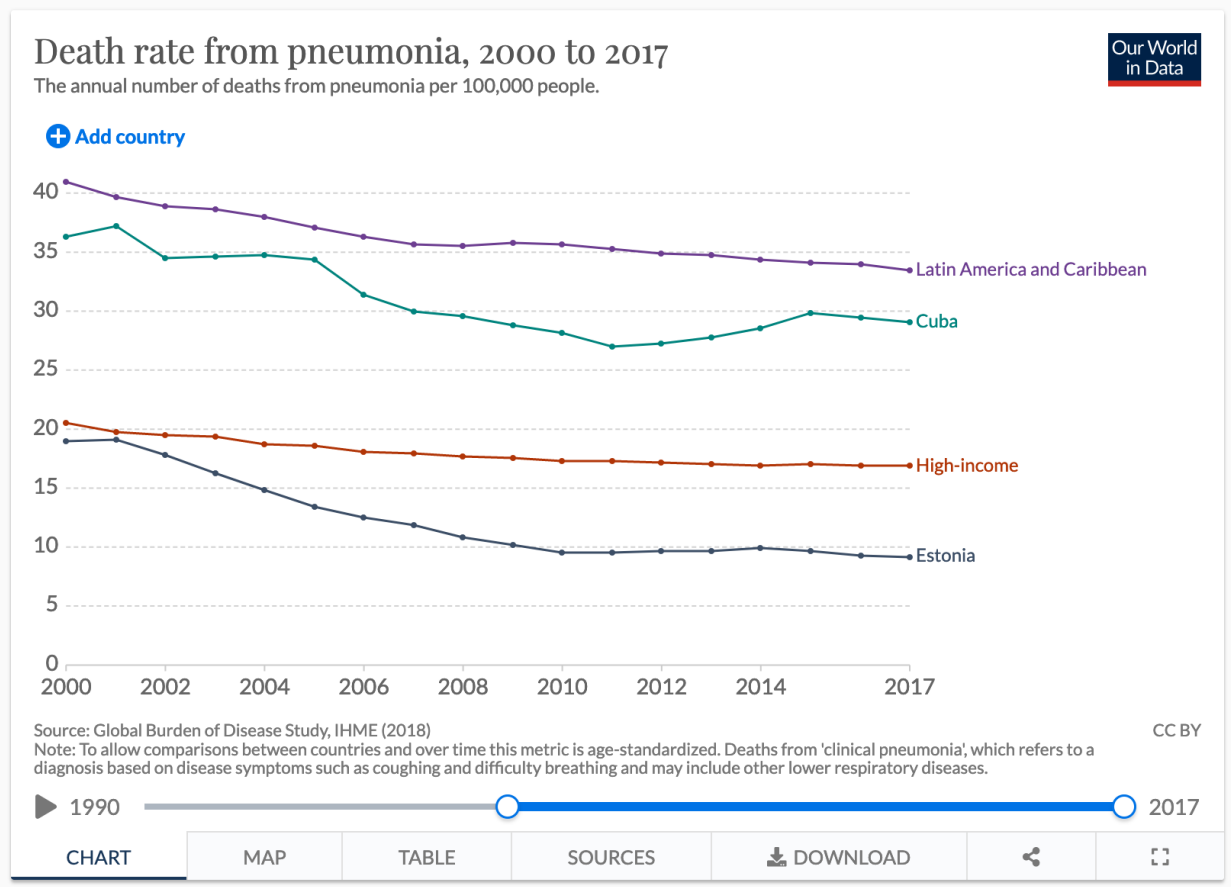
Utilizaremos inferencia Bayesiana para regularizar el problema. Seguiremos suponiendo que

$$y_i \,|\, n_i, \theta_i \sim \text{Poisson}(\lambda_i),$$

pero ahora necesitamos una distribución previa para θ_i. Sabemos, por lo anterior, que el modelo Poisson-Gamma es conjugado. Por lo tanto requerimos una distribución Gamma. Sólo falta elicitar los hiperparámetros.

No todos somos expertos en salud ni tenemos conocimiento previo. Sin embargo, podemos visitar esta página para darnos una idea de las tasas de mortalidad por neumonía en el resto del mundo.

A continuación se muestra las tasas de mortalidad para los últimos años para algunos paises y la region de America Latina y el Caribe.



Considera los siguientes puntos:

- Las tasas anteriores han sido calculadas con un método que incorpora la estructura demográfica de cada país y la estandariza con respecto a la pirámide poblacional mundial. En nuestro ejemplo, nuestras tasas no serán ajustada de tal forma (este método se conoce en inglés como *age-standarized mortality rates*).
- Las tasas reportadas tienen una base distinta, pues son reportadas con respecto a una población de 100,000 habitantes. Es decir, son tasas de mortalidad anuales para poblaciones de 100K habitantes. Por ejemplo, un valor de 5 significa que en promedio 5 habitantes por cada 100K mueren de neumonia al año.

Pregunta 9) Con esto en mente, escribe los límites necesarios para encontrar una distribución Gamma adecuada. Encuentra la solución al sistema de ecuaciones no lineales por medio de la función `nleqslv::nleqslv`. Escribe tu razonamiento para seleccionar dichos valores.

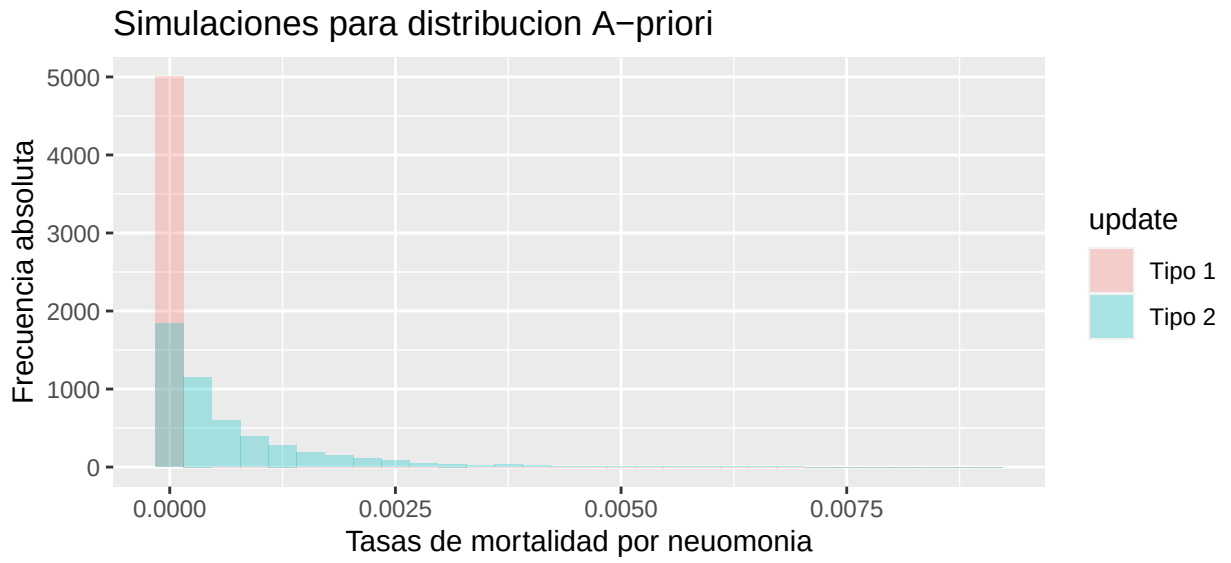
La tasa de habitantes fallecidos por neumonía (filtrado por años tomando en cuenta únicamente del año 2015 hasta el 2017) presenta un comportamiento uniforme (no en el sentido estricto, pero sí de una forma bastante aproximada) a lo largo del periodo de estudio en este proyecto para los distintos grupos de edades, por lo que, se decidió dar como cota superior el promedio de la tasa de mortalidad para el rango de edad "70+ years old", ya que son los máximos alcanzado y como límite inferior el promedio de las tasa del rango "$5-14$ years old", pues son las mínimas observadas.

```
cot_sup <- mean(c(184.31, 181.42, 178.51))
cot_min <- mean(c(1.14, 1.12, 1.12))

limits <- c(cot_min, cot_sup)/(10**5)
```

```r
gamma.limits <- function(x){
  # reparametrizamos para que el problema sea mas "fácil" en términos numéricos.
  log_alpha <- x[1]
  log_beta  <- x[2]

  # definimos las cotas de probabilidad
  p_cota <- 0.1
  c(    pgamma(limits[1], exp(log_alpha), rate = exp(log_beta)) - p_cota,
    1 - pgamma(limits[2], exp(log_alpha), rate = exp(log_beta)) - p_cota)
}

initial_guess <- c(log(1), log(1))

results <- nleqslv(initial_guess, gamma.limits)

params.prior <- exp(results$x)

rm(cot_sup, cot_min, results)
```

Pregunta 10) Grafica los histogramas de una variable aleatoria Gamma con los valores iniciales para el problema de optimización y con los finales de dicho algorimo. Esto te servirá de verificación que el método funciona adecuadamente.

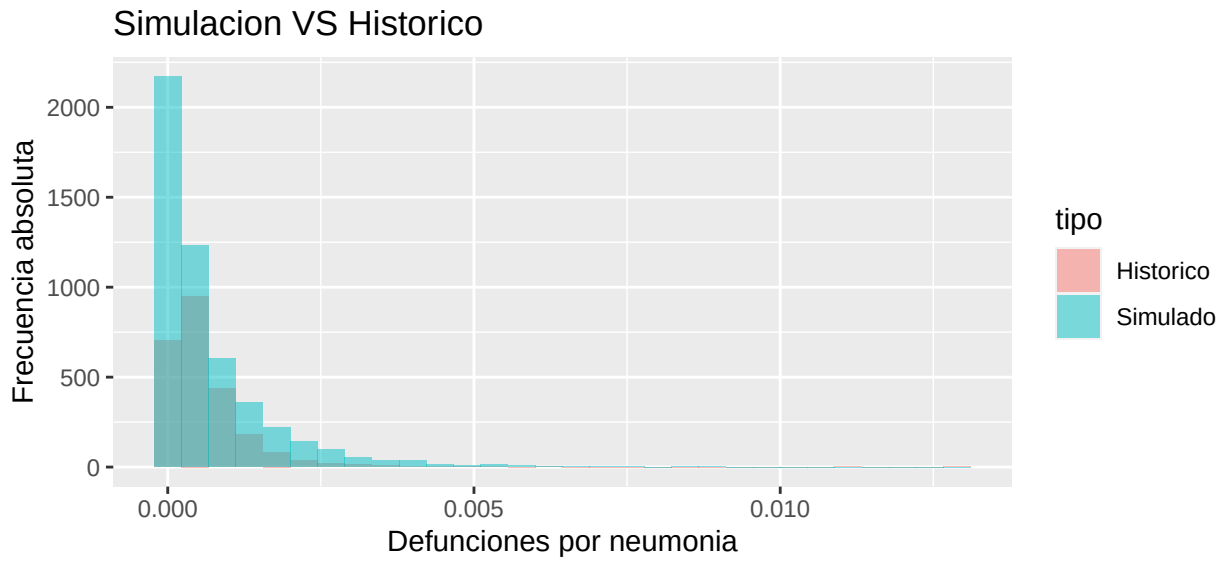
```r
set.seed(min(claves_unicas))

n_sim <- 5000
g_priori_1 <- rgamma(n_sim, initial_guess[1], initial_guess[2])
g_priori_2 <- rgamma(n_sim, params.prior[1], params.prior[2])

data_gamma <-
  data.frame(update = c( rep("Tipo 1", n_sim),
                         rep("Tipo 2",n_sim)),
  value = c(g_priori_1, g_priori_2))

data_gamma %>%
  ggplot(aes(x = value, fill = update)) +
  geom_histogram(alpha=0.3, position = 'identity', bins = 30) +
  # scale_fill_manual(values=c("#69b3a2", "#404080")) +
  xlab("Tasas de mortalidad por neuomonia") +
  ylab("Frecuencia absoluta") +
  ggtitle('Simulaciones para distribucion A-priori')
```



En comparación de las distribuciones apriori, observamos que la segunda ("Tipo 2"), tiene un mejor comportamiento en comparación con la primera, que indica que no habrá ninguna muerte con neumonía, en contraste los datos no lo indican así.

Pregunta 11) ¿Cómo se compara la distribución a priori con las tasas observadas en los municipios? Puedes utilizar histogramas para estas comparaciones. Por otro lado, no te preocupes si no se ven identicas. El punto es ver que nuestras creencias iniciales se ven coherentes.

```
data_neum <- full_data %>%
  filter(municipio != 999) %>%
  ungroup() %>%
  select(tasa_mort)

data.frame(muertes = c(data_neum$tasa_mort,
                       g_priori_2),
           tipo = c(rep('Historico', length(data_neum$tasa_mort)),
                    rep('Simulado', n_sim))) %>%
  ggplot(aes(x = muertes, fill = tipo)) +
  geom_histogram(alpha=0.5, position = 'identity') +
  xlab('Defunciones por neumonia') +
  ylab('Frecuencia absoluta') +
  ggtitle('Simulacion VS Historico')
```

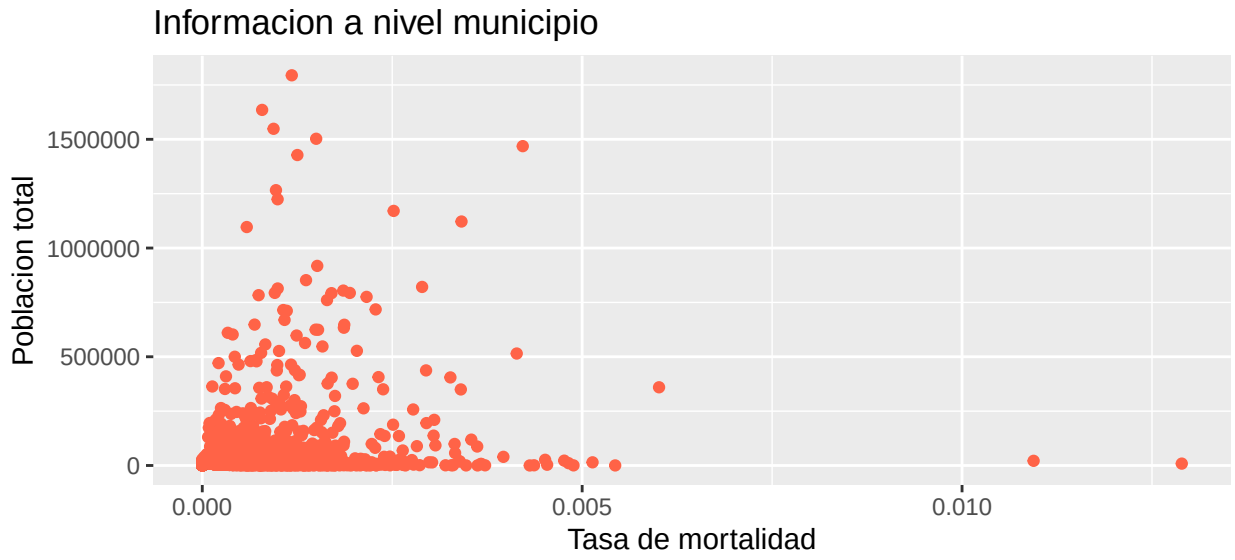



```
rm(data_gamma, data_neum, g_priori_1, g_priori_2)
```

Se sobreestima la tasa de mortalidad por neumonía para el valor 0, es decir, nuestra distribución apriori supone una gran supervivencia, lo cual no es respaldado por los datos, sin embargo, eso buscaremos cambiar incorporando la información de los datos con la distribución posterior.

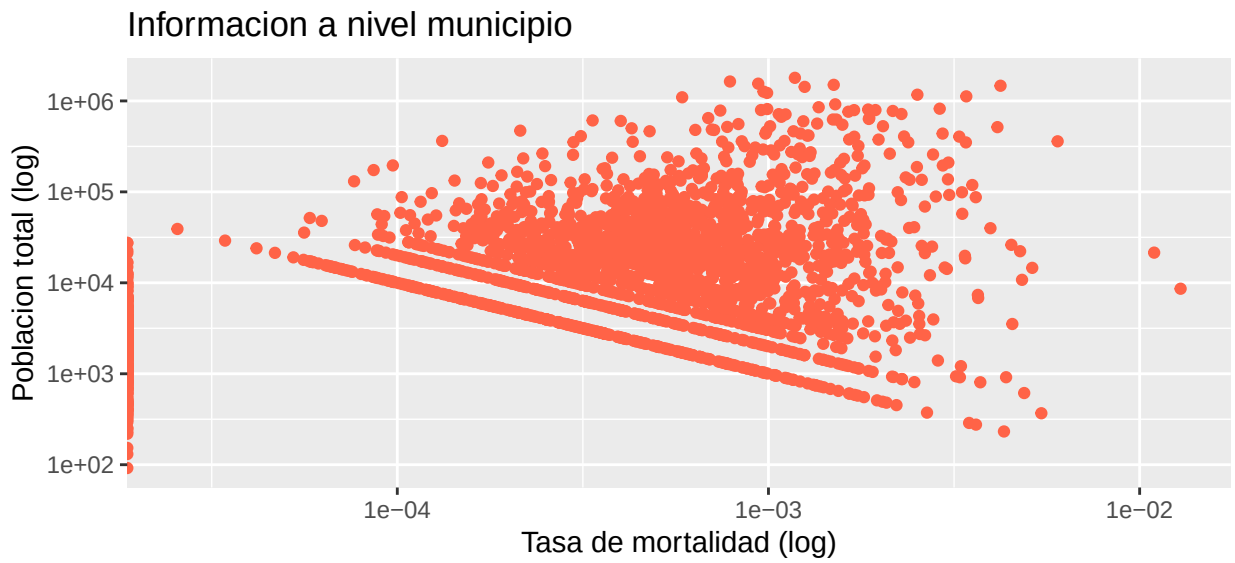
Pregunta 12) Utiliza un gráfico de dispersión para comparar las tasas observadas contra la población del municipio. ¿Qué observas? Utiliza los ejes en escala logaritmica. Para esto checa la función: `ggplot2::scale_x_log10` y `ggplot2::scale_y_log10`

```
full_data %>%
  ggplot(aes(x = tasa_mort, y = pob_tot)) +
  geom_point(color = 'tomato') +
  xlab('Tasa de mortalidad') +
  ylab('Poblacion total') +
  ggtitle('Informacion a nivel municipio')
```




```
full_data %>%
  ggplot(aes(x = tasa_mort, y = pob_tot)) +
```

```
  geom_point(color = 'tomato') +
  ggplot2::scale_x_log10() +
  ggplot2::scale_y_log10() +
  xlab('Tasa de mortalidad (log)') +
  ylab('Poblacion total (log)') +
  ggtitle('Informacion a nivel municipio')
```



El primer gráfico es un poco "difícil" (las palabras difícil, fácil, sencillo,... son demasiado relativa a las características intrínsecas de los individuos) de asimilar, pues hay una gran concentración de puntos que no permiten analizar un poco el fenómeno a similar. Sin embargo, para el segundo gráfico la historia es un distinta, ya que, pareciese que este fenómeno sigue un comportamiento un tatnto exponencial, para tasas de mortalidad bajas, la población total es alta, mientras qeu para tasas de mortalidad altas, la población es baja. Todo esto con un comportamiento exponencial. Que se presente este caso, nos da la siguiente idea intuitiva: * La neumonía no afecta de manera uniforme a la población, pues dependiendo de la estructura demográfica (en cuanto a edades) de la población, la nuemonía afectará de manera distinta a cada uno de los distintos grupos.

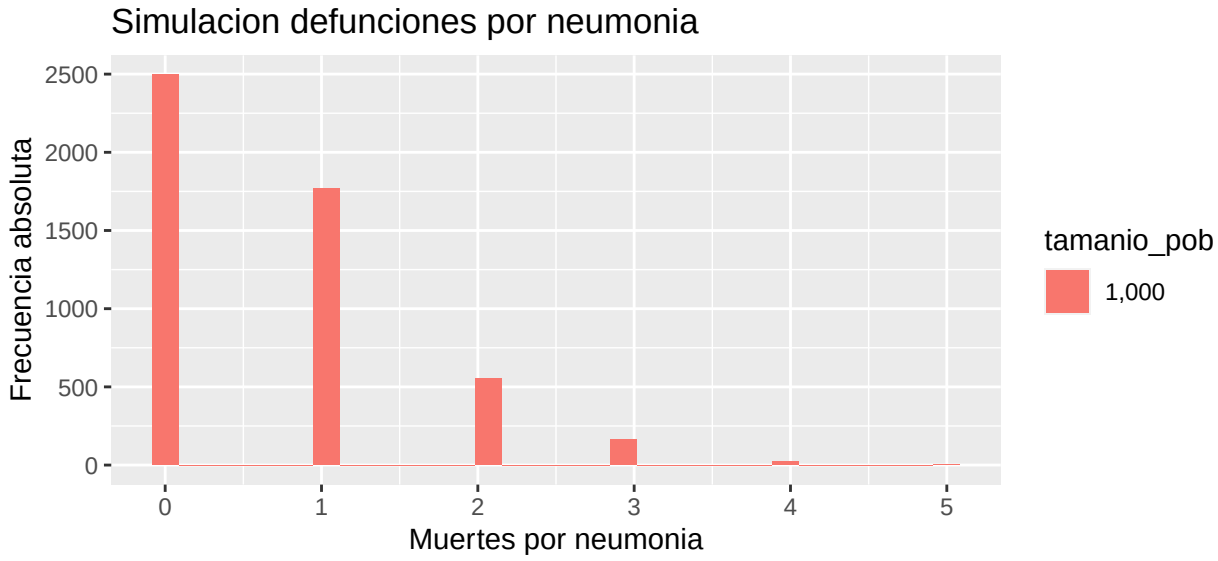
Cuando tenemos tamaños de población grande, la presencia de muertes por neumonía es alta, debido a la alta densidad poblacional y de manera inversa, cuando tenemos tamaños de población bajos, la presencia de muertes por neumonía es alta, debido a la baja densidad poblacional. Es relvante mencionar que tenemos presencia de valores atípicos en las colas, los cuales hacen que nuestra "hipótesis" sea falsa.

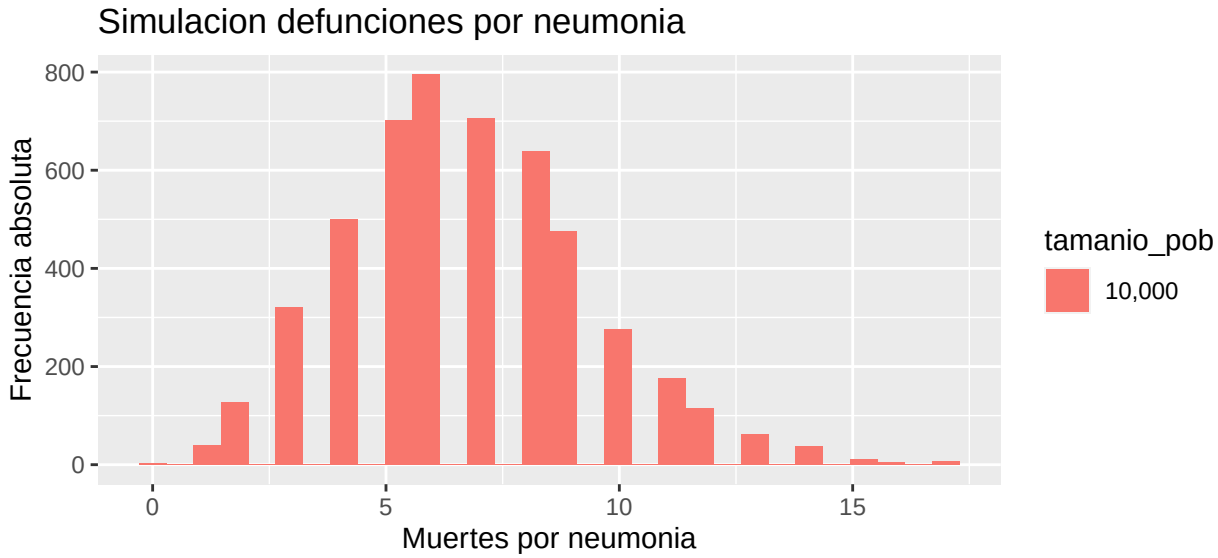
Pregunta 13) Ahora usaremos la distribución predictiva **previa** para explorar los posibles valores que tendrían los casos de muerte bajo nuestro modelo para municipios de distintos tamaños. Para este punto considera que la predictiva es una mezcla de Poisson con Gamma, como se expresa en

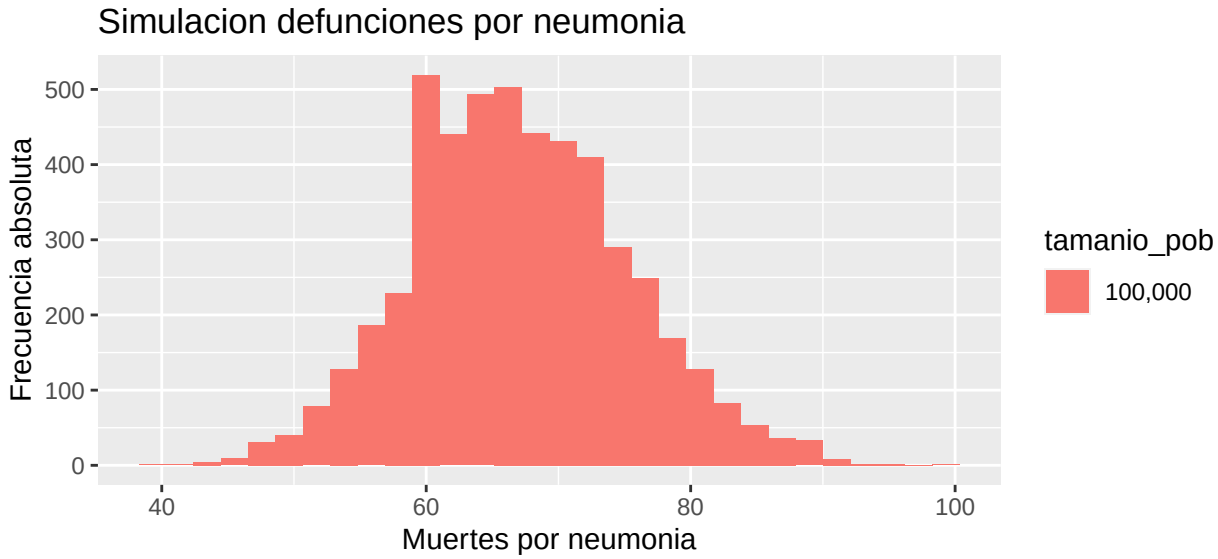
$$p(y|n) = \int \mathsf{Poisson}(y|n,\theta)\,\mathsf{Gamma}(\theta|\alpha,\beta)\,\mathrm{d}\theta.$$

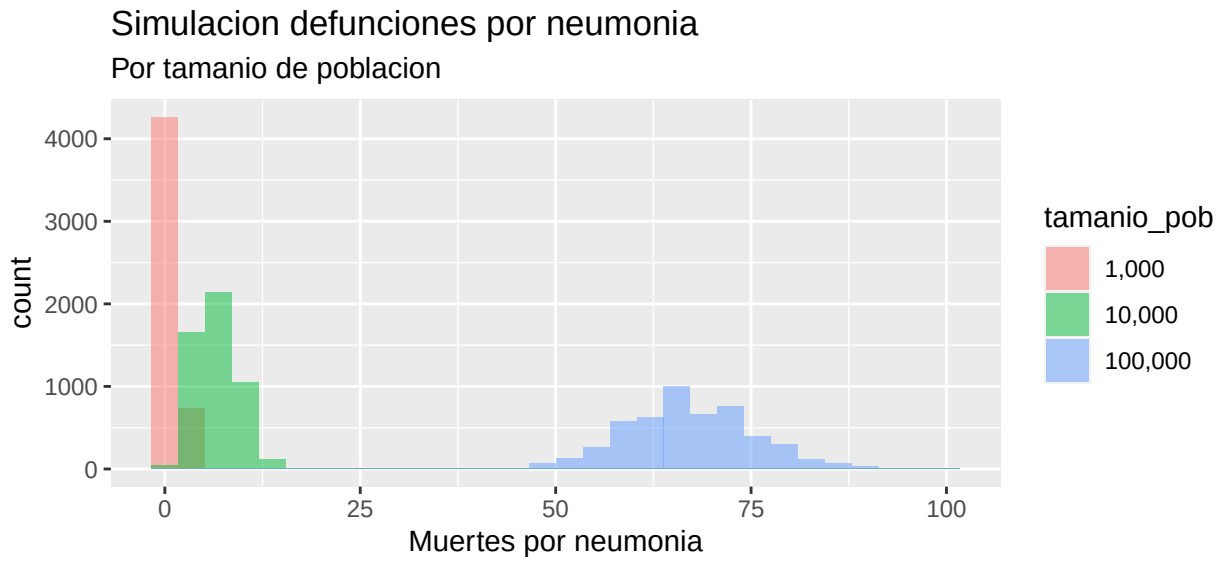
o bien, la forma en especifico de la predictiva preevia. ¡Esto ya lo has resuelto en la primera parte del examen!

Usa histogramas para ver los números de muertes en municipios hipotéticos de tamaño $n = 10^3, 10^4, 10^5$.


Simulacion defunciones por neumonia
Frecuencia absoluta
2500
2000
1500
1000
500
0
0
1
2
3
4
5
Muertes por neumonia
tamanio_pob
1,000


Simulacion defunciones por neumonia
Frecuencia absoluta
800
600
400
200
0
0
5
10
15
Muertes por neumonia
tamanio_pob
10,000


Simulacion defunciones por neumonia
Frecuencia absoluta
500
400
300
200
100
0
40
60
80
100
Muertes por neumonia
tamanio_pob
100,000




```
## $y
## [1] "Frecuencia absoluta"
##
## attr(,"class")
## [1] "labels"
```

Pregunta 14) Calcula los valores posteriores de las tasas de mortalidad bajo nuestro modelo bayesiano y compara con los estimadores de máxima verosimilitud. Para esto puedes utilizar un gráfico de dispersión cómo el visto en clase o los anteriores. ¿Observas regularización en nuestras estimaciones? ¿Qué observas si haces un grafico como el de la pregunta 12?

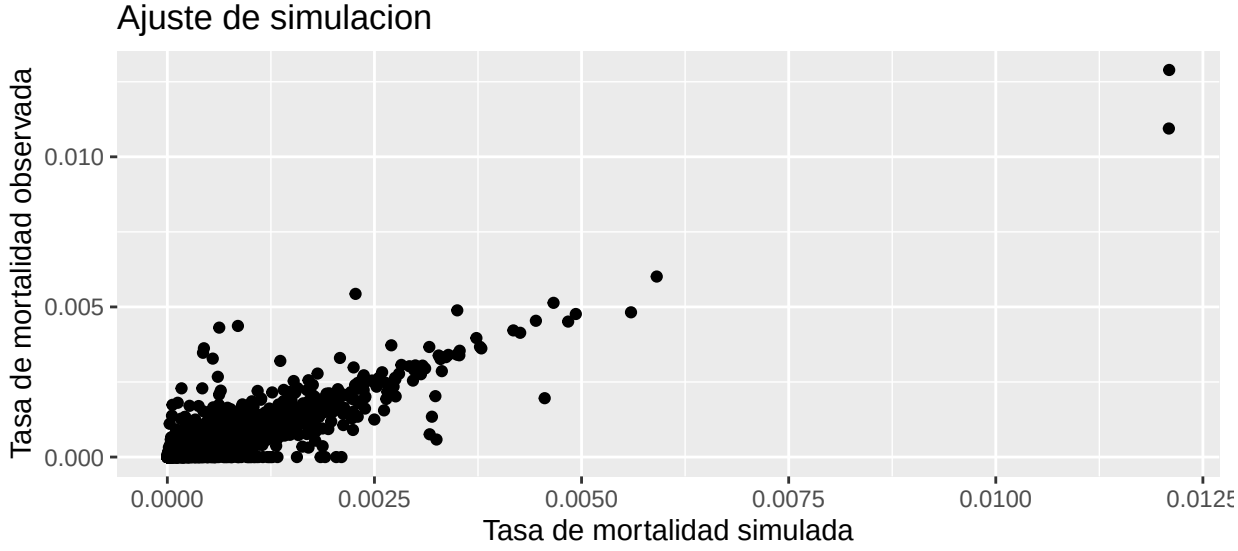
$$p(y|n) = \int \text{Poisson}(y|n,\theta)\,\text{Gamma}(\theta|\alpha,\beta)\,\mathrm{d}\theta.$$

Cabe mencionar que la distribución predictiva posterior es la siguiente:

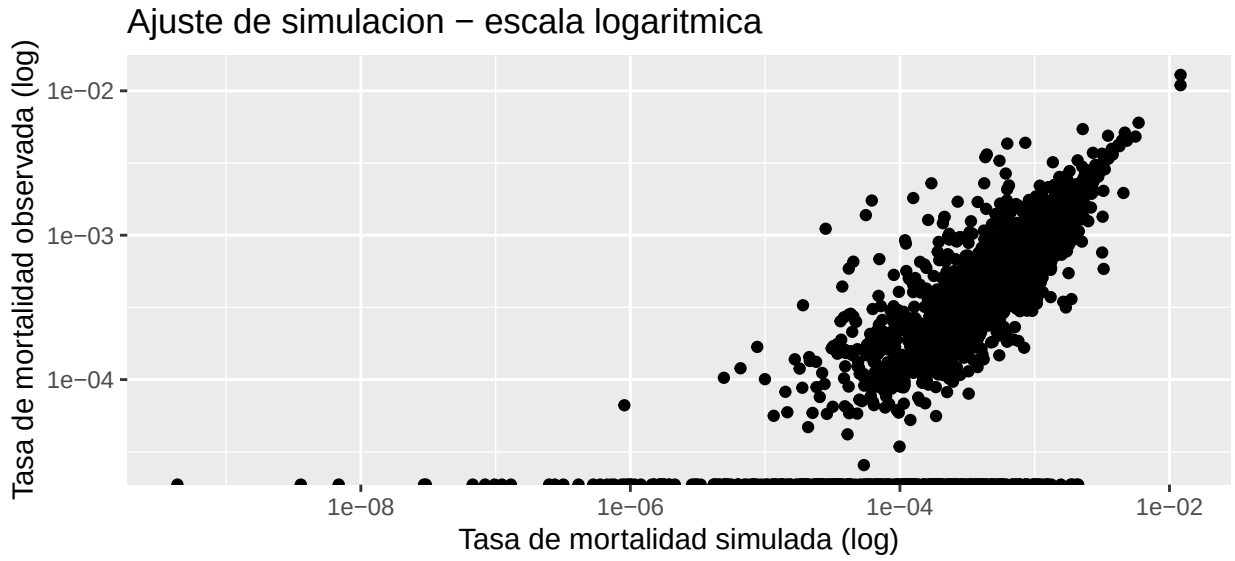
$$y|n \sim BN(\alpha\ +\ \sum_{i=1}^{n} x_i,\ \beta\ +\ n)$$

```r
set.seed(min(claves_unicas))

full_data_filt <- full_data %>%
  filter(municipio != 999)

calcular_pars_posterior <- function(num_muertes , tam_pob, params.prior){

  # Parametros iniciales
  alpha_prior <- params.prior[1]
  beta_prior <- params.prior[2]

  # Parámetros
  alpha_post <- alpha_prior + num_muertes

  # beta post
  beta_post <- beta_prior + tam_pob
  # beta_post <- beta_prior + 1
```

```
  return(c(alpha_post, beta_post))
  }

pars_post <- apply(full_data_filt[,c('muertes_neum', 'pob_tot')], 1,
             function(x) calcular_pars_posterior(x[1], x[2],
                                                 params.prior)) %>%
  t()

full_data_filt$tasa_sim_post <-  apply(pars_post, 1,
                                       function(x) rgamma(1, x[1], x[2]))

full_data_filt %>%
  ggplot(aes(x = tasa_sim_post, y = tasa_mort)) +
  geom_point() +
  xlab('Tasa de mortalidad simulada') +
  ylab('Tasa de mortalidad observada') +
  ggtitle('Ajuste de simulacion')
```




```
full_data_filt %>%
  ggplot(aes(x = tasa_sim_post, y = tasa_mort)) +
  geom_point() +
  ggplot2::scale_x_log10() +
  ggplot2::scale_y_log10() +
  xlab('Tasa de mortalidad simulada (log)') +
  ylab('Tasa de mortalidad observada (log)') +
  ggtitle('Ajuste de simulacion - escala logaritmica')
```




```
# data.frame(tasa_mort = c(full_data_filt$tasa_mort,
#                          full_data_filt$tasa_sim_post)) %>%
#   mutate(Tipo = c(rep('Observada', 2462),
#                   rep('Sim posterior', 2462))) %>%
#   ggplot(aes(x = tasa_mort, fill = Tipo)) +
#   geom_histogram(alpha = 0.5) +
#   xlab('Tasa de mortalidad') +
#   ylab('Frecuencia absoluta') +
#   ggtitle('Comparacion de simulacion')


full_data_filt$tasa_sim_ver <- full_data_filt$muertes_neum/full_data_filt$pob_tot

data.frame(tasa_mort = c(full_data_filt$tasa_sim_ver,
                         full_data_filt$tasa_sim_post)) %>%
  mutate(Tipo = c(rep('Sim Max Ver', 2456),
                  rep('Sim posterior', 2456))) %>%
  ggplot(aes(x = tasa_mort, fill = Tipo)) +
  geom_histogram(alpha = 0.5) +
  xlab('Tasa de mortalidad') +
  ylab('Frecuencia absoluta') +
  ggtitle('Comparacion de simulacion')
```

Notemos un hecho importante en la gráfica "Ajuste por simulación escala logarítmica", el cual es que para las tasas de mortalidad simuladas más altas, se vienen las tasas de mortalidad observadas más altas y de manera análoga para las más chicas. Esto es importante, pues si no fuese así, estaríamos llegando a contradicciones en el estudio. En comparación con los estimadores de máxima verosimilitud con los parámetros obtenidos por la distribución posterior, observamos que las distribuciones tienen la misma "forma", mas con el gran cambio en que para la máxima verosimilitud hay una mayor frecuencia pra las tasas más bajas a comparación de la posterior.

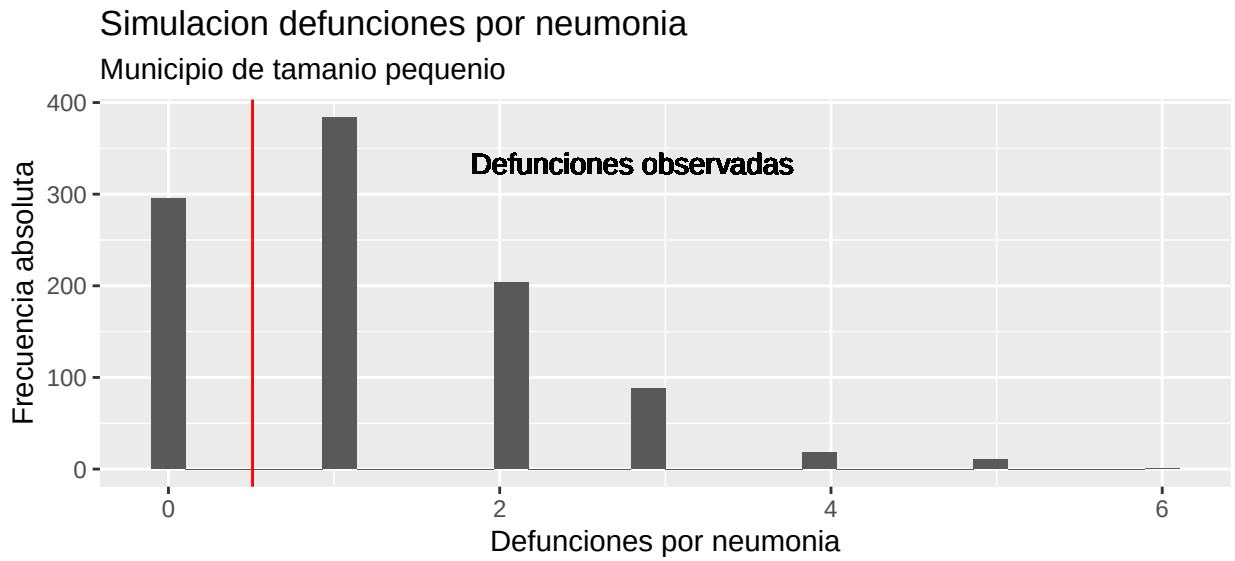
Pregunta 15) Utiliza la distribución predictiva *posterior* para verificar el ajuste del modelo. Para esto, escoge tres municipios al azar de distintos tamaños (chico, mediano, grande) Grafica un histograma y compara con el número observado de defunciones.

```
set.seed(min(claves_unicas))

full_data_filt <- full_data_filt %>%
  mutate(tamanio = ifelse(pob_tot <= 1000, 1,
                          ifelse(pob_tot <= 10000, 2, 3)))

full_data_filt %>%
  group_by(tamanio) %>%
  summarise(count = n())
```

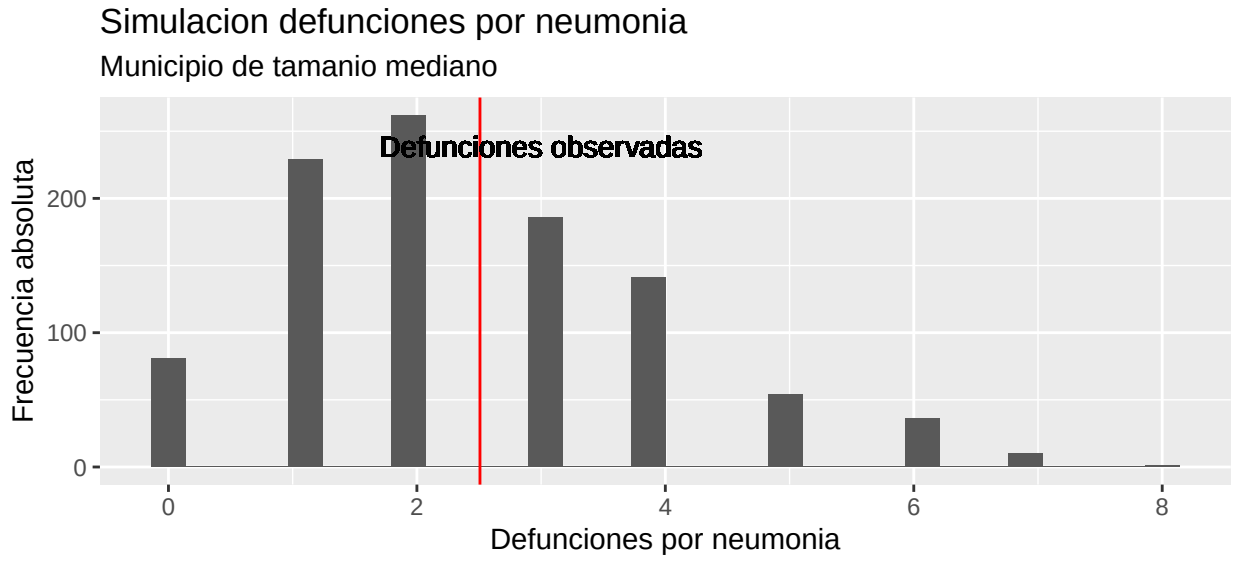
```
## # A tibble: 3 x 2
##   tamanio count
##     <dbl> <int>
## 1       1   128
## 2       2   951
## 3       3  1377
```

```
pos_chico <- sample(which(full_data_filt$tamanio == 1), 1)
pos_mediano <- sample(which(full_data_filt$tamanio == 2), 1)
pos_grande <- sample(which(full_data_filt$tamanio == 3), 1)

par_post_chico <- pars_post[pos_chico,]
par_post_mediano <- pars_post[pos_mediano,]
par_post_grande <- pars_post[pos_grande,]
```

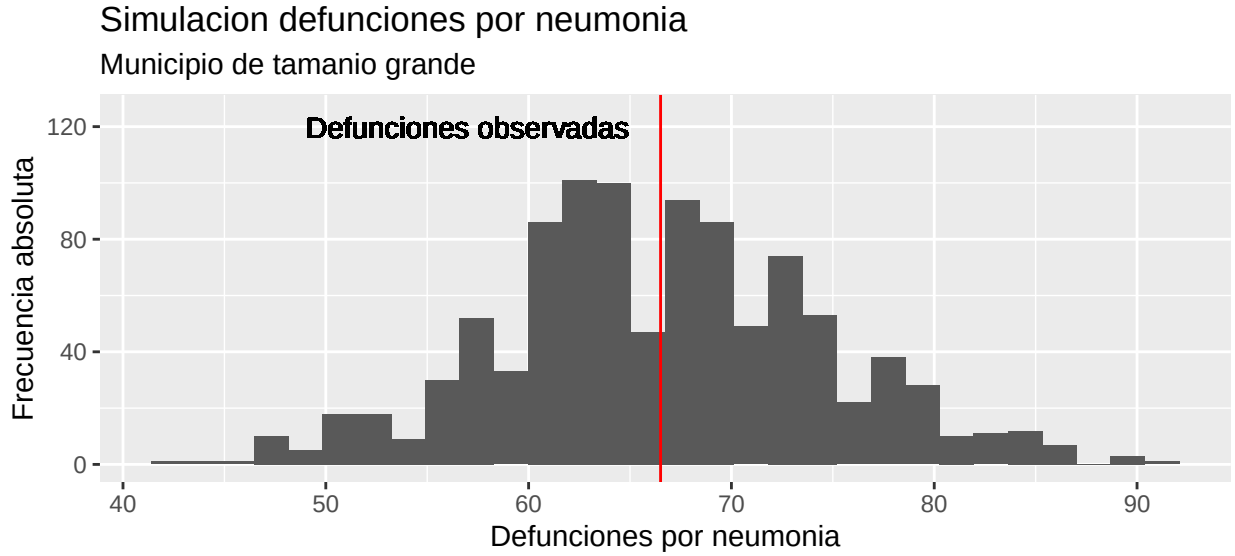
```r
data.frame(sim = rnbinom(1000,
                   size = par_post_chico[2]*(params.prior[1]) + par_post_chico[1],
                   prob = (params.prior[2] + 1)/(params.prior[2] + 1 + 1) )) %>%
  ggplot(aes(x = sim)) +
  geom_histogram() +
  geom_vline(xintercept = par_post_chico[1], color = 'red') +
  geom_text(aes(x=2.8, y=350,
                label="Defunciones observadas"),
            colour="black", vjust = 1.2) +
  xlab('Defunciones por neumonia') +
  ylab('Frecuencia absoluta') +
  ggtitle('Simulacion defunciones por neumonia') +
  labs(subtitle = 'Municipio de tamanio pequenio')
```




```r
data.frame(sim = rnbinom(1000,
                   size = par_post_mediano[2]*(par_post_mediano[1]) + par_post_mediano[1],
                   prob = (par_post_mediano[2] + 1)/(par_post_mediano[2] + 1 + 1) )) %>%
  ggplot(aes(x = sim)) +
  geom_histogram() +
  geom_vline(xintercept = par_post_mediano[1], color = 'red') +
  geom_text(aes(x=3, y=250,
                label="Defunciones observadas"),
            colour="black", vjust = 1.2) +
  xlab('Defunciones por neumonia') +
  ylab('Frecuencia absoluta') +
  ggtitle('Simulacion defunciones por neumonia') +
  labs(subtitle = 'Municipio de tamanio mediano')
```




```
data.frame(sim = rnbinom(1000,
                    size = par_post_grande[2]*(par_post_grande[1]) + par_post_grande[1],
                    prob = (par_post_grande[2] + 1)/(par_post_grande[2] + 1 + 1) )) %>%
  ggplot(aes(x = sim)) +
  geom_histogram() +
  geom_vline(xintercept = par_post_grande[1], color = 'red') +
  geom_text(aes(x=57, y=125,
                label="Defunciones observadas"),
            colour="black", vjust = 1.2) +
  xlab('Defunciones por neumonia') +
  ylab('Frecuencia absoluta') +
  ggtitle('Simulacion defunciones por neumonia') +
  labs(subtitle = 'Municipio de tamanio grande')
```



Para las simulaciones de las defunciones por neumonía de cada municipio notemos que la la cantidad de defunciones observadas se encuentra alrededor de la media de la distribución.

Incorporando Grupos de Edad

Se sabe que las muertes por neumonia no son uniformes y las tasas de mortalidad son mas altas en niños y personas mayores. Ahora realizaremos el mismo análisis considerando grupos de edad. Para esto ampliaremos nuestro modelo

$$y_{k,i} \,|\, n_{k,i}, \theta_{k,i} \sim \text{Poisson}(\lambda_{k,i}),$$

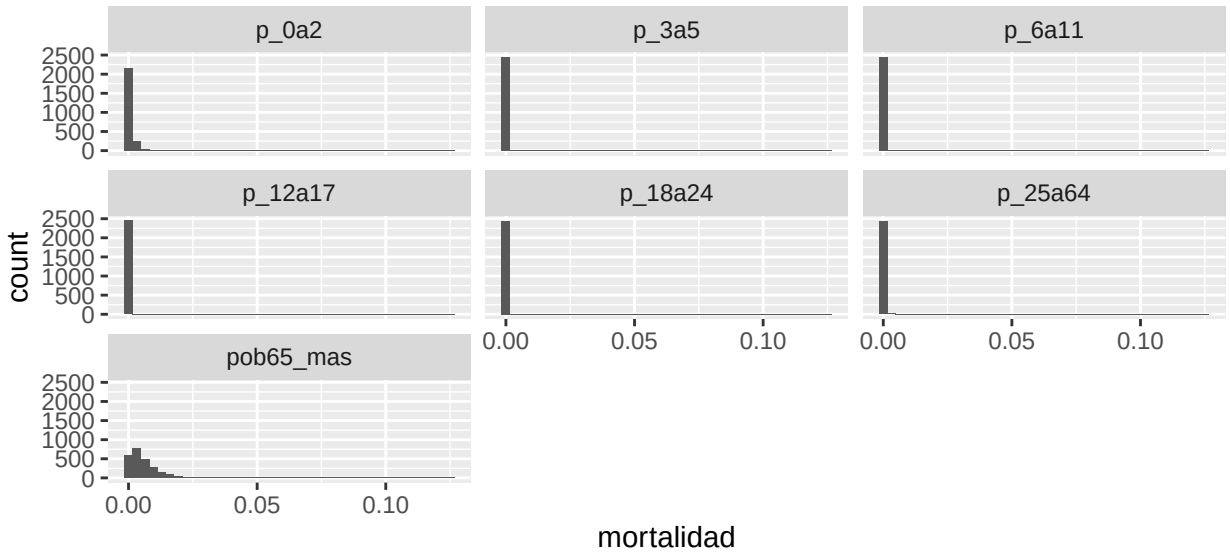
donde utilizamos el sub-indice k, i para denotar el k-ésimo grupo de edad en el i-ésimo municipio.

Pregunta 16) Genera histogramas para cada grupo de edad y discute si el supuesto anterior esta soportado por los datos. Para esto calcula las tasas de mortalidad adecuadas. Auxiliate de `ggplot2::facet_wrap`.

```
datos_edad <- poblacion %>%
    select(entidad, municipio, p_0a2:pob65_mas) %>%
  pivot_longer(p_0a2:pob65_mas,
               names_to = "GrupoEdad",
               values_to = "Poblacion") %>%
  left_join(defunciones %>%
  group_by(entidad, municipio, edad_grupos) %>%
  count(name="Defunciones") %>%
  mutate(GrupoEdad = recode(edad_grupos,
                            "[0,3)"="p_0a2",
                            "[3,6)"="p_3a5",
                            "[6,12)"="p_6a11",
                            "[12,18)"="p_12a17",
                            "[18,25)"="p_18a24",
                            "[25,64)"="p_25a64",
                            "[64,Inf]"="pob65_mas")),
  by=c('entidad', 'municipio', 'GrupoEdad' )) %>%
  mutate(Defunciones = replace_na(Defunciones, 0),
    mortalidad = Defunciones/Poblacion,
    GrupoEdad = factor(GrupoEdad,
                       levels = c("p_0a2","p_3a5","p_6a11","p_12a17",
                                  "p_18a24","p_25a64","pob65_mas")))
```

Ahora la grafica

```
datos_edad %>%
  ggplot() +
  geom_histogram(aes(x=mortalidad), bins=40) +
  facet_wrap(vars(GrupoEdad))
```




```
datos_edad %>%
  group_by(GrupoEdad) %>%
  summarise(Poblacion = sum(Poblacion))
```

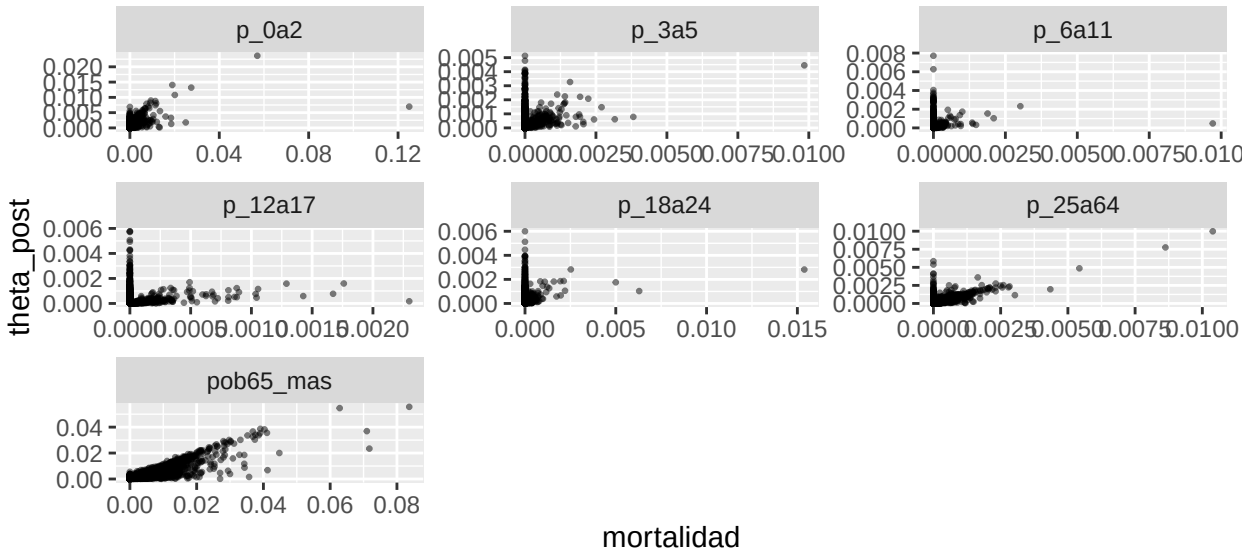
```
## # A tibble: 7 x 2
##   GrupoEdad Poblacion
##   <fct>         <dbl>
## 1 p_0a2       6157867
## 2 p_3a5       6535234
## 3 p_6a11     13318563
## 4 p_12a17    13215080
## 5 p_18a24    14207435
## 6 p_25a64    50566040
## 7 pob65_mas   6938913
```

Por motivos de simplicidad usaremos la misma distribución previa para cada tasa de mortalidad asociada a grupos de edad y municipio que en los puntos anteriores.

Pregunta 17) Utiliza graficos de dispersión para determinar si hay efectos de regularización. Las ideas las encuentras arriba en la pregunta 12 y 14.

```
vec_gamma <- Vectorize(rgamma)
datos_edad <-  datos_edad %>%
  mutate(alpha_post = params.prior[1] + Defunciones,
         beta_post = params.prior[2] + Poblacion,
         theta_post = vec_gamma(1,alpha_post,beta_post))
```

```
datos_edad %>%
  ggplot() +
  geom_point(aes(x=mortalidad, y = theta_post), size=0.5, alpha=0.5) +
  facet_wrap(vars(GrupoEdad), scale="free")
```

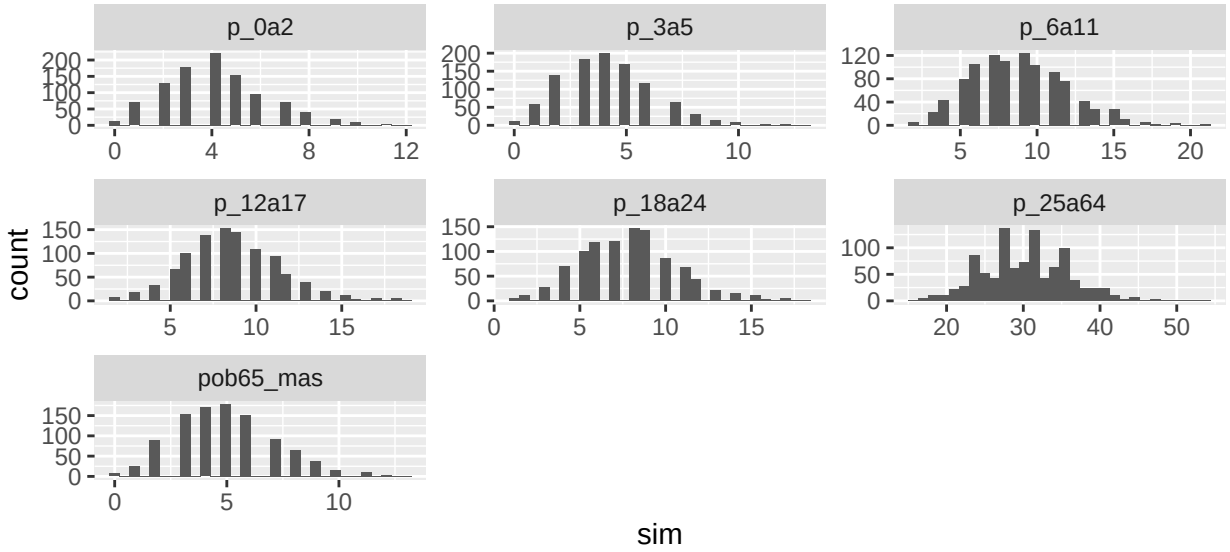


Pregunta 18) Para uno de los tres municipios que escogiste anteriormente utiliza la distribución predictiva posterior para verificar el ajuste del modelo para los grupos de edad: $[0, 3)$, $[18, 25)$ y $[64, \infty)$.

```
pos_chico <- sample(which(full_data_filt$tamanio == 1), 1)
pos_mediano <- sample(which(full_data_filt$tamanio == 2), 1)
pos_grande <- sample(which(full_data_filt$tamanio == 3), 1)

chico <- paste0(full_data[pos_chico,'entidad'],
                full_data[pos_chico,'municipio'])
mediano <- paste0(full_data[pos_mediano,'entidad'],
                  full_data[pos_mediano,'municipio'])
grande <- paste0(full_data[pos_grande,'entidad'],
                 full_data[pos_grande,'municipio'])

muestra_edad <- datos_edad %>%
  mutate(region = paste0(entidad, municipio)) %>%
  filter(region == chico | region==mediano | region==grande) %>%
  mutate(tamano = factor(ifelse(region==chico,
                          "chico",
                          ifelse(region==mediano, "mediano", "grande")),
                          levels = c("chico", "mediano", "grande")))

res = muestra_edad

for (i in 1:1000){
  res = rbind(res, muestra_edad)
}
  vec_bn <- Vectorize(rnbinom)


res <- res %>%
  mutate(sim = vec_bn(1,  size = Poblacion*(params.prior[1])+Defunciones,
                          prob = (params.prior[2]+1)/(params.prior[2]+1+1)))
```

```
res %>%
  filter(tamano == "grande") %>%
  ggplot() +
```

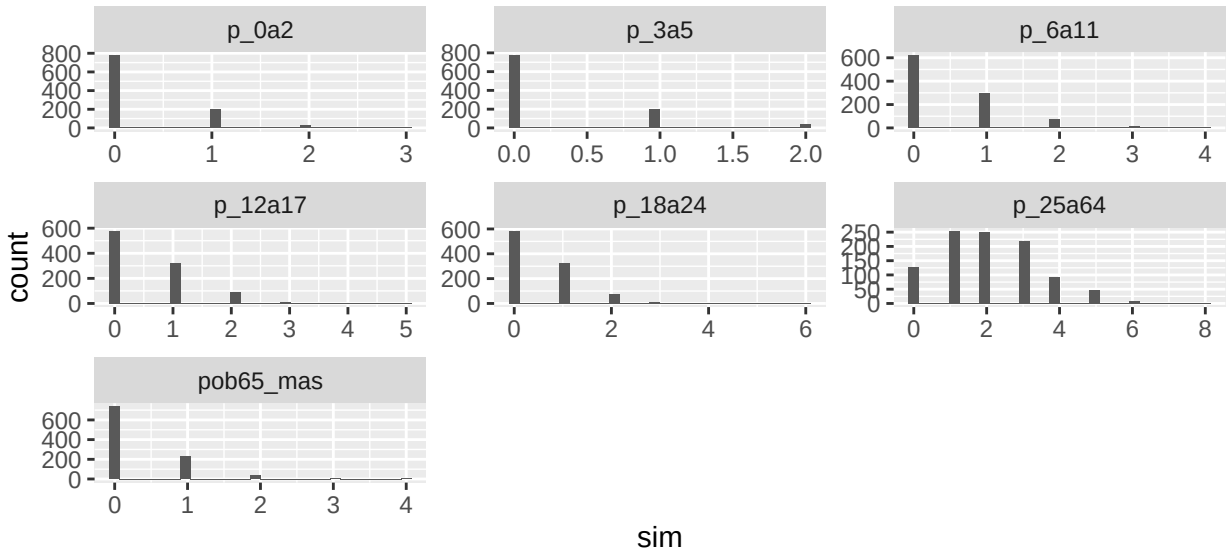
```
  geom_histogram(aes(x=sim)) +
  facet_wrap(~GrupoEdad, scale = "free")
```




```
muestra_edad %>% filter(tamano=="grande")
```

```
## # A tibble: 7 x 12
##   entidad municipio GrupoEdad Poblacion edad_grupos Defunciones mortalidad
##   <chr>   <chr>     <fct>         <dbl> <chr>              <dbl>      <dbl>
## 1 05      033       p_0a2          6150 <NA>                   0  0
## 2 05      033       p_3a5          6289 <NA>                   0  0
## 3 05      033       p_6a11        13034 <NA>                   0  0
## 4 05      033       p_12a17       12782 <NA>                   0  0
## 5 05      033       p_18a24       11756 <NA>                   0  0
## 6 05      033       p_25a64       45220 [25,64)                4  0.0000885
## 7 05      033       pob65_mas      7305 [64,Inf]              52  0.00712
## # ... with 5 more variables: alpha_post <dbl>, beta_post <dbl>,
## #   theta_post <dbl>, region <chr>, tamano <fct>
```

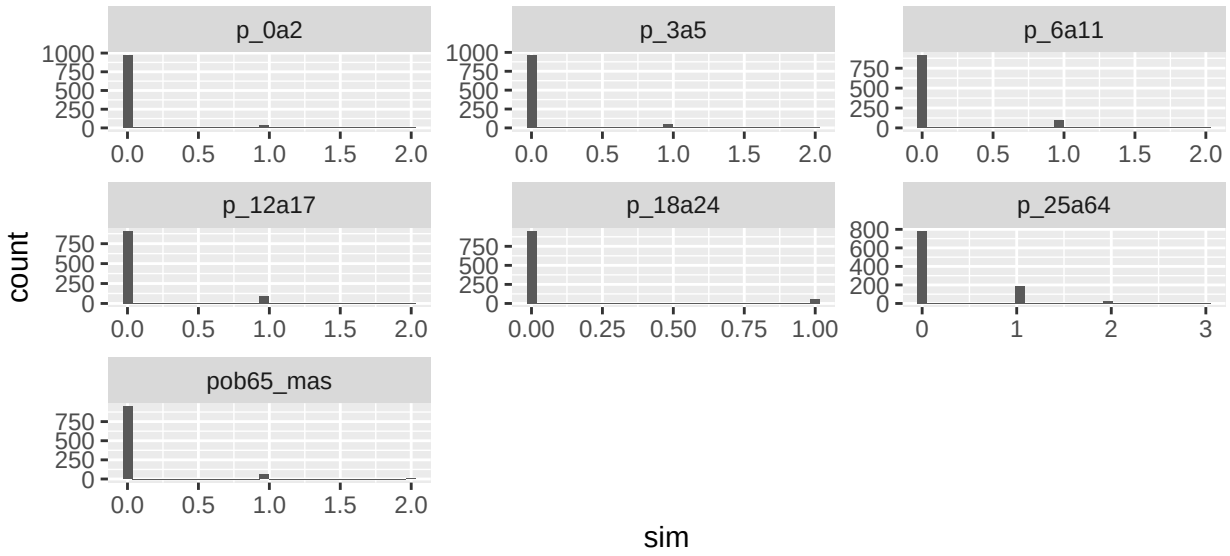
```
res %>%
  filter(tamano == "mediano") %>%
  ggplot() +
  geom_histogram(aes(x=sim)) +
  facet_wrap(~GrupoEdad, scale = "free")
```




```
muestra_edad %>% filter(tamano=="mediano")
```

```
## # A tibble: 7 x 12
##   entidad municipio GrupoEdad Poblacion edad_grupos Defunciones mortalidad
##   <chr>   <chr>     <fct>         <dbl> <chr>              <dbl>      <dbl>
## 1 30      074       p_0a2           378 [0,3)                  1    0.00265
## 2 30      074       p_3a5           363 <NA>                   0    0
## 3 30      074       p_6a11          806 <NA>                   0    0
## 4 30      074       p_12a17         762 <NA>                   0    0
## 5 30      074       p_18a24         802 <NA>                   0    0
## 6 30      074       p_25a64        3130 <NA>                   0    0
## 7 30      074       pob65_mas       458 [64,Inf]               4    0.00873
## # ... with 5 more variables: alpha_post <dbl>, beta_post <dbl>,
## #   theta_post <dbl>, region <chr>, tamano <fct>
```

```
res %>%
  filter(tamano == "chico") %>%
  ggplot() +
  geom_histogram(aes(x=sim)) +
  facet_wrap(~GrupoEdad, scale = "free")
```




```
muestra_edad %>% filter(tamano=="chico")
```

```
## # A tibble: 7 x 12
##   entidad municipio GrupoEdad Poblacion edad_grupos Defunciones mortalidad
##   <chr>   <chr>     <fct>         <dbl> <chr>              <dbl>      <dbl>
## 1 20      168       p_0a2            55 <NA>                   0          0
## 2 20      168       p_3a5            55 <NA>                   0          0
## 3 20      168       p_6a11          151 <NA>                   0          0
## 4 20      168       p_12a17         170 <NA>                   0          0
## 5 20      168       p_18a24          96 <NA>                   0          0
## 6 20      168       p_25a64         367 <NA>                   0          0
## 7 20      168       pob65_mas        83 <NA>                   0          0
## # ... with 5 more variables: alpha_post <dbl>, beta_post <dbl>,
## #   theta_post <dbl>, region <chr>, tamano <fct>
```

Conclusiones:

¡El último modelo (edad-municipio) incorpora alrededor de 17K parámetros distintos! Sin duda, no es parsimonioso. De hecho, este modelo representa el extremo en complejidad para esta situación. Podemos incorporar una estructura jerárquica donde podemos interpretar una estructura multi-nivel en cuanto al conocimiento que podemos generar. Esto es por que en la estuctura de dependencia dejamos la misma distribucion previa sin importar municipio o grupo de edad. En cursos posteriores exploraremos estas opciones. Pero ahora, ¡a descansar!